TEC-0120

# Representation, Modeling, and Recognition of Outdoor Scenes

Martin A. Fischler
Robert C. Bolles

SRI International
333 Ravenswood Avenue
Menlo Park, CA 94025-3493

May 1999

Prepared for:
Defense Advanced Research Projects Agency
3701 North Fairfax Drive
Arlington, VA 22203-1714

Monitored by:
U.S. Army Corps of Engineers
Topographic Engineering Center
7701 Telegraph Road
Alexandria, Virginia 22315-3864

US Army Corps
of Engineers
Topographic
Engineering Center

TEC

U.S. ARMY TOPOGRAPHIC ENGINEERING CENTER
TEC

Destroy this report when no longer needed.
Do not return it to the originator.

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

The citation in this report of trade names of commercially available products does not constitute official endorsement or approval of the use of such products.

# REPORT DOCUMENTATION PAGE

*Form Approved*
*OMB No. 0704-0188*

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
May 1999

3. REPORT TYPE AND DATES COVERED
Combined Quarterly/Annual January 1997 - April 1998

4. TITLE AND SUBTITLE

Representation, Modeling, and Recognition of Outdoor Scenes

5. FUNDING NUMBERS

DACA76-92-C-0008

6. AUTHOR(S)

Martin A. Fischler and Robert C. Bolles

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

SRI International
333 Ravenswood Avenue
Menlo Park, CA 94025-3493

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Defense Advanced Research Projects Agency
3701 N. Fairfax Drive, Arlington, VA 22203

U.S. Army Topographic Engineering Center
7701 Telegraph Road, Alexandria, VA 22315-3864

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

TEC-0120

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

The goal of this project is to advance the state-of-the-art in scene interpretation for autonomous systems that operate in natural terrain. In particular, techniques are being developed for representing knowledge about complex cultural and natural environments so that a computer vision system can successfully plan, navigate, recognize, and manipulate objects, and answer questions or make decisions relevant to this knowledge. The results to date include the development of new representations and techniques for rapidly modeling terrain from multiple images, and for the recognition and reliable labeling of such scene attributes and components as color, texture, shadows, and a variety of linear structures (skyline, ridgelines, road, etc.). The most recent results are detailed in three papers included as appendices to this report.

14. SUBJECT TERMS

Machine vision, automated scene analysis, object recognition, terrain modeling, linear delineation

15. NUMBER OF PAGES
50

16. PRICE CODE

| | | | |
|---|---|---|---|
| 17. SECURITY CLASSIFICATION OF REPORT | 18. SECURITY CLASSIFICATION OF THIS PAGE | 19. SECURITY CLASSIFICATION OF ABSTRACT | 20. LIMITATION OF ABSTRACT |
| UNCLASSIFIED | UNCLASSIFIED | UNCLASSIFIED | UNLIMITED |

# TABLE OF CONTENTS

PREFACE....................................................................................v

1.0 OBJECTIVE..........................................................................1

2.0 APPROACH...........................................................................1

3.0 PROGRESS............................................................................1

4.0 SUMMARY OF RECENT ACCOMPLISHMENTS AND ACTIVITIES....3

BIBLIOGRAPHY.........................................................................8

LIST OF APPENDICES..............................................................10

APPENDIX A, "Finding the Perceptually Obvious Path"..........................11

APPENDIX B, "Robotic Vision: Sketching Natural Scenes"......................12

APPENDIX C, "Sketching Natural Terrain from Uncalibrated Imagery" .....13

# Glossary of Acronyms

| | |
|---|---|
| CONDOR: | Proper-name |
| HUB: | Proper-name |
| APGD: | Automatic Population of Geospatial Databases |
| BOS: | Proper-name (derived from: Battlefield Observer System) |
| RADIUS: | Proper-name (derived from: Research and Development for Image Understanding Systems) |
| RCVW: | Rapid Construction of Virtual Worlds |
| UGV: | Unmanned Ground Vehicle (DARPA) Program |
| RCDE: | Radius Common Development Environment |
| NIMA: | National Imagery and Mapping Agency |

# PREFACE

This research is sponsored by the Defense Advanced Research Projects Agency (DARPA), and monitored by the U.S. Army Topographic Engineering Center (TEC) under contract DACA76-92-C-0008, titled "Representation, Modeling, and Recognition of Outdoor Scenes." The DARPA Program Manager is **Mr. George Lukes**, and the TEC Contracting Officer's Representative is **Ms. Lauretta Williams**.

## 1.0 OBJECTIVE

The primary goal of this project is to advance the state-of-the-art in scene interpretation for autonomous systems that operate in natural terrain. In particular, techniques are being developed for representing knowledge about complex cultural and natural environments so a computer vision system can successfully plan, navigate, recognize, manipulate objects, answer questions, or make decisions relevant to this knowledge.

## 2.0 APPROACH

This work integrates our continuing advances in four separate technologies to achieve the goal of providing a foundation for the design of highly competent machine vision systems capable of autonomous operation in the outdoor world:

- First, stored knowledge (such as geospatial data and object models, as well as contextual dependencies and inter-relationships) is used to overcome inherent weaknesses in the best "self-contained" image-analysis algorithms. This approach is reflected in the prior SRI development of the CONDOR[1] and HUB[2] systems, and the current APGD[3]/BOS[4] architecture that relies on context, function, and purpose, as well as visually observed geometric shape, to recognize scene objects.

- Second, significant progress has been made in developing compact and expressive representations for modeling, and ultimately recognizing, objects encountered in the natural world. Computational efficiency, thus real time performance, is critically dependent on using effective representations for both reference models and sensed data.

- Third, global optimization techniques are being developed that require reasonable amounts of computation but produce results not obtainable by local analysis methods. This work has been applied to building volumetric models of objects detected in range data and stereo pairs, as well as for delineation, partitioning, and feature extraction in single images.

- Fourth, techniques are being developed that are able to simultaneously, or incrementally, exploit multiple views of a scene in compiling a complete scene model. The SRI's previously developed Epi-polar plane image analysis technique and the current work on deformable mesh representations are examples of how multi-image collections can be used to construct a geometric scene model that is superior to a sequence of independent stereo reconstructions.

## 3.0 PROGRESS

Theoretical and practical progress has been made in recovering scene models from physical constraints, stored knowledge, multiple views, and image sequencesor collections. The

[1] CONDOR: Proper –name (see Glossary of Acronyms page IV, for a definition).

[2] HUB: Proper name.

[3] APGD: Automatic Population of Geospatial Databases.

[4] BOS: Proper name (derived from: Battlefield Observer System).

results in the area of geometric recovery are centered on novel deformable mesh and particle representations for describing scene surfaces at a level of organization higher than that of the conventional dense array of depths, and on an associated continuation-type optimization method for rapidly modeling terrain geometry in terms of these mesh representations. The deformable mesh also serves as a substrate for consistent integration of terrain and surface features (e.g., rivers, drainage, and roads); we are able to refine several models simultaneously and enforce geometric and semantic constraints between the objects and the terrain. In tests using this new technique, we have repeatedly been able to significantly improve the accuracy of recovered terrain and object models beyond that of the best available stereo methods (e.g., P. Fua "Fast, Accurate and Consistent Modeling of Drainage and Surrounding Terrain," IJCV 26(3): 215-234, March 1998).

Methods were also developed to build simplified geometric descriptions of the terrain, using the triangulated meshes. A mesh is first overlaid on a stereo disparity image such that no facets cross discontinuities, then it is reduced in complexity to minimize the number of triangles in the mesh. The technique developed for this is similar to "mesh decimation" used in graphics, however, constant reference is made to the initial disparity image, thus ensuring that the final mesh is still an accurate representation of the disparity image. Data reductions of over 98 percent of the data in the original mesh has been achieved. When this mesh is transformed into real-world coordinates we have a simplified yet accurate triangulated description of the visible surfaces. A detailed discussion of the earlier work on geometric recovery can be found in the third yearly report (April 1995) for this project.

Earlier in this project, a significant new advance in the long-standing problem of duplicating human performance in recovering 3-D models of terrain and man-made objects from qualitative and imprecise line drawings (e.g., of terrain elevations presented in an approximate and uncalibrated contour map, or building edges as in a single approximate projection of the corresponding wire-frame) was made. This work can greatly simplify communication problems between man and machine in such applications as robotic mission planning and construction of databases for use in robotic navigation. A paper describing this work has been published ("An optimization based approach to the interpretation of single line drawings as 3-D wire frames," IJCV 9(2): 113-136, Nov 1992). On-going related work in "sketching" the geometry of natural scenes has led to (new) additional results of both theoretical and practical importance; some of these new results, still being further developed and evaluated, are presented in an appendix to this report (T. Luong "Sketching Natural Terrain from Uncalibrated Imagery").

The problem of automatically recognizing objects appearing in images of the outdoor world has proven to be extremely difficult because of the lack of explicit shape models. Most computer-based recognition techniques rely on explicit knowledge of shape, but rocks, trees, and other natural objects cannot be successfully described in this way; even such generic man-made objects as roads, bridges, and buildings are more likely to satisfy functional constraints rather than being exemplars of some generic geometric blueprint. It is necessary to replace explicit shape with a more general way of describing natural objects and complex man-made structures. The proposed approach is based on employing a small set of techniques that can very reliably organize the "pixel-level" image data as a basis for higher level analysis. Finding the appropriate combination of low-level data-description and associated extraction techniques is a key problem and a primary concern in this project.

Two of the techniques that have emerged from the work in this area meet the criterion of generality and robustness. The first is a generic way to find candidate line structure in an image (see following discussion). The second is a way to organize such data into perceptually coherent and semantically meaningful units. In a recent paper (IEEE-PAMI 1994) we described our progress in the design of a partitioning technique that is extremely robust in accomplishing the perceptual organization task and also describe how these two techniques can be applied to the problem of road delineation in aerial images.

In papers presented at the 1994 and 1996 Image Understanding Workshops, we described our work on the detection and extraction of linear features in imaged data -- one of the most useful and effective of our core scene-analysis techniques: We use the minimum spanning tree and a new "network" structure we devised as the primary representations. Semantic constraints control the tree/network construction thus establish the universe of possible paths (both in our data structures and in the image being analyzed). We define the characteristics of the linear structures we are looking for as attributes of the branches in the tree/network and provide computationally effective methods for finding paths that maximize scores for the desired attributes. Filtering techniques, parameterized by context evaluation procedures (or externally provided information) operate at a number of decision points in the optimization process and in final acceptance of the selected path(s). We have implemented specialized experimental versions of the generic delineation technique to recognize various types of extended terrain features and navigation obstacles including the skyline, ridgelines, trees, roads, and paths. The problem of finding linear features in aerial images has been of special interest, and as discussed later, has resulted in a major advance in automating the task of modeling roads in the compilation of geospatial databases.

We recently assembled a system, based on the above two scene analysis techniques, delineation and partitioning, to demonstrate the feasibility of automatically modeling an urban building complex from Interferometric Synthetic Aperture Radar (IFSAR) data. The approach was to first smooth and partition the IFSAR data using minimal description length encoding and generalized connected-components analysis. The "footprints" of the individual buildings were then found by using the curve partitioning algorithm (IEEE-PAMI 1994) to decompose the boundary of each isolated object (connected component) into a rectilinear set of edges. Finally, a containing 3-D shell for each building using the extracted footprint and the building height as measured in the original IFSAR data was constructed. This work demonstrates the utility and generality of the scene analysis tools we have devised.

More than 15 papers have been published describing the above work. Algorithmic techniques developed in this program have been integrated into a commercial cartographic modeling system, and used in the RADIUS[5], RCVW[6], UGV[7] and APGD programs.

After the completion of the base funding and development period of this contract, our efforts focused on improving the performance and scope of our natural/outdoor object recognition techniques. Work on recognizing complex natural and man-made objects (e.g. roads, trees, rocks, and terrain features) is based on a set of ideas and techniques we are developing for recognizing complete scene contexts, rather than instances of independent object models. We have been able to experimentally demonstrate the validity of the approach by recognizing and delineating scene objects that cannot be dealt with by conventional methods and are currently integrating the component technology into a complete testable demonstration system; this work is described in greater detail below and in three appendices.

[5] RADIUS: Proper name (derived from: Research and Development for Image Understanding Systems).

[6] RCVW: Rapid Construction of Virtual Worlds.

[7] UGV: Unmanned Ground Vehicle (DARPA) Program.

# 4.0 SUMMARY OF RECENT ACCOMPLISHMENTS AND ACTIVITY

The contribution of our current work concerned with natural scene description and recognition is a set of computer algorithms capable of using one or more images of an outdoor scene to create a labeled scene sketch that makes explicit the qualitative geometry and identifies the major visible natural and man-made objects. A key problem (noted earlier) is the necessity to replace reliance on a generally unavailable explicit shape with more general ways of recognizing and describing natural objects and complex man-made structures.

The approach is to first select (or define) a smaller set of primitive yet pervasive features that can reliably be extracted from most images of natural scenes. This set of primitives (currently consisting of: color, texture, shadows, depth, surface orientation, and linear structures) are combined to identify clear instances of the natural objects of interest using a "production rule" type paradigm [ref: Strat and Fischler, "Context-based vision," IEEE PAMI, Oct 91]. By using these recognized objects as exemplars, we can invoke a nearest-neighbor statistical classifier to label other, possibly less obvious, instances of the objects we are looking for. Objects of interest (the semantic vocabulary) include rocks, trees, brush, grass, water, snow, ground, sky, ridgelines, holes/ditches, roads, paths, fences, poles, cliffs, ground-plane, and shadows.

The key ideas underlying this work are:

1. Models are described by objective functions referenced to some appropriate representation; feature extraction is accomplished by finding image structures for which the relevant objective function is optimized. We generally require that the representations we construct be suitable directly viewable replacements (with respect to the given interpretation task) for the original image but require only a small fraction of the original data storage; finding such "reduced representations" is a an important step in the solution process.

2. Recognition-technique selection and corresponding parameter settings are based on context and confirmed by "built-in" self-evaluation functions.

3. Selection and intense development to produce a few highly refined and reliable "core" techniques as the base for implementing a much broader class of feature recognition/extraction methods.

Much of this work is now focused on the development of algorithms for the recognition of a wide range of natural/outdoor objects of importance to APGD, robotic navigation, and outdoor scene modeling. We have devised a way to quickly convert a color image into a labeled Scene Sketch that can be directly used for these purposes. The Scene Sketch (described in more detail in an Appendix to this report) is currently a composition of the four implemented components: color-sketch, shadow-sketch, line-sketch, and texture-sketch. An additional component being developed is a depth/terrain-profile sketch (this work is described in more detail in an enclosed appendix).

The Color Sketch. We have implemented a set of color-based classification algorithms, using classical feature-space partitioning techniques and decision rules based on the physics of outdoor-scene color-image formation to produce a semantic description of a scene in terms of the categories: sky/clouds/snow, live vegetation, earth/ground/rocks, water/rocks, and shadows/unknown. This list, which can obviously be expanded, appears to be within the competence of the very fast and simple pixel-level color-image processing algorithm we employ given that we allow the somewhat intermixed categories; e.g., a volcanic rock can look quite dark or "ground-like," while under suitable lighting and view conditions, (some portion of) a rock composed of granite will reflect light just like the surface of a body of water -- and is indistinguishable from water just using local color.

The Shadow Sketch is primarily used as an indicator of raised objects -- the ground surface does not cast a shadow nor is it typically self-occluding. The algorithm we use exploits the nominal intensity ordering (from dark to light): shadows, raised vegetation, ground-level vegetation, ground/earth, water, sky. The critical observation (regularity) we exploit is that almost any black and white image can be reduced to a one-bit depiction of a scene, by almost any reasonably low threshold, and still allow a human observer to correctly interpret the qualitative scene geometry. A smaller threshold will cause some potential obstacles to be missed; a higher threshold will mark some darker, but unraised regions as obstacles.

The Texture Sketch attempts to use the shading and texture variations in a black and white image to infer the orientation of the visible scene surfaces. The algorithm exploits two regularities: raised (relatively isolated) objects have significant vertical edge content; and because of "foreshortening," most of the projectively imaged edges on a relatively horizontal plane in a typical outdoor scene will appear to be horizontal in the image. Rather than explicitly extracting edges and attempting to analyze them, we are currently attempting to identify isolated raised objects and horizontal (support) surfaces by finding coherent regions where the local intensity gradient is predominantly vertical or horizontal.

The Line-Sketch makes explicit physical and geometric discontinuities in material-type, illumination, depth (i.e., occlusion boundaries), and surface orientation. In spite of the seeming simplicity of the task, fully automated robust methods for constructing a Line-Sketch do not (yet) exist. Our work on constructing a Line-Sketch as a component of the Scene Sketch is currently focused on finding roads, paths, skylines, and ridge-lines. We have a simple but effective algorithm for finding individual skyline points given the availability of the Color-Sketch. After first removing small isolated clusters of "sky-points," we mark the location of the lowest (smallest y-coordinate) sky-point in each column of the image array. While explicit linking is not necessary for the Line-Sketch, it is needed to properly extend the skyline through occlusions (which, in turn, is necessary for a simple technique to detect raised vegetation). We invoke the linking techniques presented in [Fischler94, Fischler96] for this purpose. Work on extending the Line-Sketch to include ridgelines is nearing completion.

An important application of the work on generic linear delineation has been its extension to the problem of fully automating the task of finding roads in aerial imagery -- a key component of our DARPA-sponsored work concerned with the APGD. The linear delineation algorithms developed in this project formed the core of a fully automated system for road extraction in aerial imagery of rural scenes and has resulted in what we believe is the most competent facility available for this purpose.

The Depth/Terrain-Profile Sketch has recently been implemented, but we have not yet fully evaluated its performance (see enclosed appendix). The approach selected requires only a pair of uncalibrated images, but it produces a sketch where the order with respect to the dimensions of height above the ground plane and depth are correct. In addition, a dense representation is generated as a set of profile lines. One key idea in this work is to be able to use multiple images taken from arbitrary locations, or even from unknown sources, in constructing this 3-D sketch. The stereo approach to 3-D recovery requires two images taken close together in both space and time and involves determining the disparity for each point along an epipolar line. The epi-polar line is determined only by the geometry of the cameras, and within this line, each point corresponds to a different depth. By contrast, given a fixed depth, we propose to find all the points that lie at this depth. This is based on the idea that for the points that lies at a fixed depth; there is an analytical relation between their projections in multiple images. These points form a curve, called profile line, which is the trace of the terrain surface on a plane in 3-D at the given depth. By sweeping the 3-D space with a set of different depths, a representation of the terrain is obtained. Our work provides a new approach to qualitative terrain modeling; it show that in order to generate a qualitatively useful elevation map, full calibration of the cameras is not necessary; instead, the only new requirement is the identification of correspondences on the horizon line.

While there are important reasons to look beyond conventional stereo as a way of modeling the geometry of natural terrain, stereo-based geometric recovery is still the most direct approach to dealing with this problem. Never the less, stereo matching often fails in ground-level vegetated environments: occlusions, discontinuous surfaces, uncertain lighting, non-Lambertian reflection, extreme disparity ranges, homogeneous surfaces and regions (e.g., snow, water, sky) are some of the problems that must be addressed. Recent progress has been made in attempting to construct an algorithm capable of human-level stereo-matching performance for the natural/outdoor world. The approach is based on the requirement for explicitly categorizing and searching for a set of 5-10 known error sources, and for exploiting independent information sources in judging correspondence correctness. In particular, image photometry, imaging geometry, and scene semantics are able to provide three independent "opinions" on the validity of a proposed correspondence. A report describing our work in this area is in preparation.

Integration. Techniques based on the Scene Sketch and associated algorithmic techniques being developed to explicitly recognize the more important and prominent terrain features and navigation obstacles. For example, by first solving the simpler problem of delineating the skyline in a color image, we are then able to choose a region above the skyline as an exemplar of "sky" for use by the color classifier. Texture, shadows, and shape will allow us to find a few obvious instances of vegetation and rocks as additional exemplars for the color classifier. We can now label most of the scene using the color classifier, and then, (at least partially) check the result for semantic consistency: The pixels labeled sky by the color classifier should all exist above the skyline found by the linear delineation process; if the skyline is interrupted by a nearby (as measured by our depth measuring techniques) thin raised object, the object should be labeled as a tree or a pole. A relatively horizontal/flat region, depending on its color, should be grass, dirt, water, or rock etc.

We have crossed a threshold in the development and integration of the scene sketching algorithms where there is sufficient content to permit meaningful test and evaluation. We have assembled a database of approximately 50 images (150 color-component images); these pictures come from a variety of sources, but primarily the foothills near Stanford University, California deserts, and the California Sierra's at high elevations (above 8,000 feet). The algorithms that produce the Color-Sketch, the Shadow-Sketch, and the Line-Sketch provide good results although they can be improved. The texture sketch we currently compute is more problematic; it often provides good results that are valuable and not easily obtained from other sources, but it not as reliable as we would desire. We are working on an improved algorithm for constructing this component of the scene sketch.

The Color Sketch, because it semantically labels the scene content, is the most important and useful component of the complete Scene Sketch. In our experiments, we have found that when we can reliably locate the sky-regions and skyline (in the imagery being processed), we can use the implied constraints to both extend, and more reliably detect, the semantic categories depicted in the Color Sketch. Therefore, the algorithms that are employed to find the sky and skyline have been improved, and are being further refined, as they appear to be a key to any further advance in improving and extending the Scene Sketch.

Additional aspects of the above work are described in three appendices to this report; (a) M.A. Fischler, "Finding the Perceptually Obvious Path," (b) M.A. Fischler, "Robotic Vision: Sketching Natural Scenes, (c) Q.T. Luong, "Sketching Natural Terrain from Uncalibrated Imagery. Two papers currently being prepared for the 1998 IU Workshop will describe work on Road and Street delineation in aerial imagery (based on our fundamental work on generic linear delineation), and in automated highly robust correspondence/matching of point features in multiple images (with application to stereo depth recovery).

In summary, the major recent accomplishments for this effort are:

1. Work on fully automated linear-delineation in aerial imagery has resulted in what we believe is the most competent algorithm available for this purpose. This algorithm has been transferred to and installed in the RCDE[8] for testing and use in our Defense Advanced Research Projects Agency (DARPA)/NIMA[9] APGD program.

2. A set of algorithms for recognizing objects appearing in color photographs of natural outdoor scenes and for recovering scene geometry without requiring camera calibration or stereo correspondence has been developed. This on-going work can already deal with recognition problems beyond the competence of any other known technique.

3. An approach for achieving human-level accuracy in establishing stereo correspondences has been developed. The method still remains to be tested, but we have demonstrated error-free performance in our most recent experiments.

The current goals are:

1. Complete the work in automated delineation of linear structures by devising adaptive self-tuning and self-evaluation procedures.

2. Complete the work on recognition of objects in color images of natural outdoor scenes and evaluate the performance of the algorithms.

3. Complete the development and evaluate our algorithm for (essentially) error free stereo matching.

[8] RCDE: Radius Common Development Environment.

[9] NIMA: National Imagery and Mapping Agency.

# BIBLIOGRAPHY

Fua P. "Fast, Accurate and Consistent Modeling of Drainage and Surrounding Terrain," IJCV 26(3): 215-234, March 1998.

Fua P. and Y.G. Leclerc, "Combining stereo, shading, and geometric constraints for surface reconstruction from multiple views," Technical Conference Geometric Methods in Computer Vision II of SPIE Symposium, San Diego, CA, July, 1993

Fua P. and Y. G. Leclerc "Object-Centered Surface Reconstruction: Combining Multi Image Stereo and Shading," IJCV, 1994.

Fua P. and Y.G. Leclerc, "Using 3-Dimensional Meshes to Combine Image-Based and Geometry-Based Constraints," ECCV, Stockholm, Sweden, May 1994.

Fua P. and Y.G. Leclerc, "Registration without Correspondences," CVPR, Seattle, Washington, pp. 121-128, June 1994.

Fua P. and Y.G. Leclerc, "A Unified Framework to Recover 3-D Surfaces by Combining Image-Based and Externally-Supplied Constraints," Proc. DARPA Image Understanding Workshop, Monterey, CA, November 1994.

Fua P. and Y.G. Leclerc, "Image Registration without Explicit Point Correspondences," Proc. DARPA Image Understanding Workshop, Monterey, CA, November 1994.

Fua P., "Surface Reconstruction Using 3-D Meshes And Particle Systems," Third International Workshop on High Precision Navigation, Stuttgart, Germany, April, 1995

Fua P., "Reconstructing Complex Surfaces from Multiple Stereo Views," (Submitted to ICCV, Boston, MA, June 1995).

Fischler  M.A. and H.C. Wolf, "Saliency detection and partitioning planar curves," Proc. Image Understanding Workshop, Washington D.C., pp. 917-931, April 1993.

Fischler  M.A. and H.C. Wolf "Locating perceptually salient points on planar curves," IEEE-PAMI, vol. 16 (2):1-17, February 1994.

Fischler M.A., "The Perception of Linear Structure: A Generic Linker," Proc. DARPA Image Understanding Workshop, Monterey, CA, November 1994.

Fischler M.A., "Robotic Vision: Sketching Natural Scenes," Proc. DARPA Image Understanding Workshop, Palm Springs, CA, February 1996.

Fischler M.A., "Finding the Perceptually Obvious Path," DARPA Image Understanding Workshop, May 1997.

Leclerc Y.G. and M.A. Fischler, "An Optimization-Based Approach to the Interpretation of Single Line Drawings as 3-D Wire Frames," Int. J. Computer Vision, 9 (2):113-136, 1992.

Luong T., "Sketching Natural Terrain from Uncalibrated Imagery," DARPA Image Understanding Workshop, May 1997.

# BIBLIOGRAPHY (Continued)

Neuenschwander W., P. Fua, G. Szekely, and O. Kubler, "Initializing Snakes," CVPR, Seattle, WA, June 1994.

Neuenschwander W., P. Fua, G. Szekely, and O. Kubler, "Using Boundary Conditions to Improve Snake Convergence," ICPR, Jerusalem, Israel, October 1994.

Strat T.M. and M.A.Fischler, "The Role of Context in Computer Vision," Proc. Workshop on Context-Based Vision, Cambridge, Mass., June 1995.

# APPENDICES

Fischler M.A., "Finding the Perceptually Obvious Path," DARPA Image Understanding Workshop, May 1997.

Fischler M.A., "Robotic Vision: Sketching Natural Scenes," Proc. DARPA Image Understanding Workshop, Palm Springs, Calif., February 1996.

Luong T., "Sketching Natural Terrain from Uncalibrated Imagery," DARPA Image Understanding Workshop, May 1997 (Updated Version).

# APPENDIX A:

## "Finding the Perceptually Obvious Path"

M.A. Fischler

ARPA Image Understanding Workshop, May 1997

# Finding the Perceptually Obvious Path *

**Martin A. Fischler**
Artificial Intelligence Center, SRI International
333 Ravenswood Ave., Menlo Park, CA 94025 USA
E-MAIL: fischler@ai.sri.com

## Abstract

This paper is primarily concerned with the problem of finding a single perceptually obvious path (POP) in an image; e.g., an isolated road in an overhead view of a desert scene, or a particular line, drawn on a piece of paper, that a person points at. We briefly describe those relevant parts of a system designed to address the general problem of automatically delineating line-like structures, but focus on the perceptual, semantic, and computational issues relevant to this particular problem.

## 1 Introduction

This paper is primarily concerned with the problem of finding a single perceptually obvious path (POP) in an image (or selected image window); e.g., an isolated road in an overhead view of a desert scene, or a particular line, drawn on a piece of paper, that a person points at.

In reference [5] we described an architecture (Figure 1: LD block diagram) that we have found to be both general and effective for addressing the delineation problem; it involves the following subsystems and processes:

(a) Detector/Binarizer Subsystem. Binarization of the gray-level image retaining the perceptual saliency of the linear structures (e.g., Figure 2b or 3b);

(b) Generic Linear Delineation Subsystem. Partitioning and linking the binary markers into a collection of independent (perceptually obvious) generic open paths (e.g., Figure 1c, 2c).

(c) Semantic Linear Delineation Subsystem. Splitting, semantic filtering, and relinking the generic (perceptually salient) paths to obtain semantically significant delineations. Our goal here could be to find a collection of independent paths (open, closed, or both), a linked network (with or without explicit path extraction), or to find a single "best" path.

We briefly describe relevant parts of the above system, but focus on the perceptual, semantic, and computational issues relevant to this particular problem, especially the Semantic Delineation Subsystem and our proposed solution for the final process – relinking a subset of the filtered line segments into a single POP.

## 2 Overall Rational and Main Problems to be Solved

Given two curves, we typically have no precise quantitative procedure for determining which of the two would be more perceptually salient to a normal human observer. By perceptually

*This work was sponsored by the Defense Advanced Research Projects Agency under contract DACA76-92-C-0008 monitored by the U.S. Army Topographic Engineering Center, Alexandria, VA. The author also acknowledges the many contributions made by Helen Wolf to the work presented in this paper. The views and conclusions contained in this document are those of the author and should not be interpreted as representing the official policies, either expressed or implied, of the Advanced Research Projects Agency, the United States Government, or SRI International.

salient we mean that, after a very brief inspection, one alternative would be chosen over the other as depicting the presence of some interesting/important natural or man-made feature, or coherent structure, in an image of a natural scene; or likely to be found first if both were judged to be coherent; or judged to be a better exemplar of some semantic category. The available ranking criterion for generic curves is largely limited to the qualitative Gestalt laws of perceptual organization [1] – proximity, closure, simplicity, similarity, good continuation (smoothness), and symmetry. In addition we might have some semantic or physical constraints that could be used to disqualify a curve from being a member of a target semantic category. For example, a curve that "doubled-back" on itself (i.e., was multi-valued for azimuth) would typically not be a valid skyline for an image taken with a horizontally held camera; or, an isolated closed curve in an aerial image would be unlikely to depict an interstate freeway. In general, at this time, we can only be expected to make gross judgments in our ranking – e.g., to find a best (perceptually salient) curve when there is really only one viable candidate in the search area.

Because of occlusions and background structure, there generally is no simple way to partition the image into curves, associated with coherent objects, that are complete and have no contamination by extraneous background content. If we tried to list or assemble all possible curves prior to ranking them, an image with as few as 20-40 curve-points would be computationally impractical to process because of the factorial growth in the possible number of curves. What is implied by the above considerations is that a single step solution to the problem of selecting a single most salient curve is probably not attainable; we must perform a sequence of grouping, filtering, and information-reduction steps to eliminate unlikely candidates as early in the selection process as possible, and then make our final selection on a greatly simplified reduction of the originally presented data.

We have examined two distinct approaches to the delineation problem in general, and to finding the POP in particular: (a) Dynamic Programming (DP) [2] which is capable of finding a least-cost path in a real-valued 2-D array (which could be the original picture, or some derived overlay called a "cost" image), and (b) a number of graph-theoretic techniques which, in practice, require an early binarization of the input image.

DP, or any other global optimization technique that can operate on the actual input data becomes computationally infeasible for anything other than cost/objective functions that are very "local" in nature. I.E., the cost of a path going through a particular pixel in an image should only be a function of an attribute list attached to that pixel and (say) the cost of appending the given pixel to a path that passes through an adjacent pixel – rather than being dependent on (say) the specific positioning of the previous five pixels in the curve segment to which attachment is being considered. Thus, the nominal generality of full global optimization is not really attainable because of computational considerations. Even if we could contend with the computational difficulties, there is the further problem of actually specifying the global cost/objective function that approximately models human perceptual behavior in interpreting graylevel images – this is an even more difficult unsolved problem.

In the approach we will now discuss, we have found (through a combination of theory and experiment – but this is primarily an empirical result) that it is possible to automatically construct a binary overlay, of almost any non-contrived graylevel image, that will retain the perceptual saliency of the linear structures (paths). It is further the case that it is now (in a binary image) possible to define the primary cues that underlie our perception of a line or path: relative proximity and smoothness of the binary (1 or 0) pixels defining the line/path. Although not a traditional Gestalt property, persistence (e.g., coherent path length) is also cue of major importance; the other Gestalt cues play a (sometimes dominant) role only when there is ambiguity due to contending interpretations, or when we recognize some known shape or repeated structure.

Generic (perceptual rather than application dependent) clustering and linking are effectively (but not perfectly) achieved by employing a modified Minimum Spanning Tree (MST) algorithm with a bound on inter-point distance. The MST algorithm we devised for this purpose can be made to run in time proportional to the number of points being processed (because the points are represented by bounded integer coordinates, their density is not arbitrary).

The result of the above steps is a collection of disjoint MST's which can be separately parsed to to provide a collection of line-segments (RPATHS) as the final output of the generic linking component of our system. This parsing process involves (1) finding a primary path through the tree (typically a diameter path), (2) trimming-back branches with ragged ends, (3) pruning short branches, (4) partitioning the remaining collection of branches into disjoint paths which are pair-wise linked at the MST nodes according to geometric and (original-image) intensity smoothness criterion. An example showing the result of this process is presented in Figures 2c and 3c.

## 2.1 On the Combinatorics of Finding A Perceptually Obvious Path

Assume that we start with a binarized image depicting a single POP. If we had a criterion function (CF) that allowed us to rank alternative POP candidates, we observe that the naive solution of generating and ranking all possible paths is computationally infeasible for any realistic problem. Since there are n! possible paths on n points, and $20! > 10^{18}$, a problem with as few as 20 points would be impossible to solve this way.

In general, we must address two sub-problems: (1) selecting/partitioning the actual path-points from the set of potential path-points, and (2) sequencing the selected path-points. Let us assume that we are given the points that actually constitute the solution (POP). A very reasonable CF, based on the primary Gestalt property of proximity, is density (number-of-path-points/path-length); i.e., we want to find the shortest path that contains all the given/selected points. What we have just established is that a simplification (sub-problem) of our original problem is the Traveling Salesman Problem (TSP) if the POP is closed, or the problem of finding a "Messenger" (open) path. Both the TSP and the "Messenger" path problem are known to be computationally intractable for large values of n (NP-hard). For example, (at least) until recently, the largest value of n for which there is a known solution to a non-contrived TSP was 318 cities [7][6]. While there are fast methods for finding an approximation to the solution of a Eucledian TSP problem, the perceptual character of such a solution is uncertain.

It is clear that in order to solve the POP problem we must strictly limit the the number of points that can be arbitrarily sequenced, or we must limit the number of choices that are the possible successors of any given point, or use some combination of the two preceding constraints. In a variety of problems domains that we have been concerned with (e.g., finding roads in aerial images, recognizing trees and/or finding the skyline in natural ground-level scenes), we have observed that we can usually find very dense path segments that are longer than some minimal length (related to visual detection criterion), and place perceptual and/or application-domain-related constraints on linking possibilities for these dense segments. To the extent that most of the path-points are already sequenced as members of the detected segments, and it is only the segments that must be sequenced, and even here there are only a few linking alternatives for each of the segments, we can solve the POP problem even though it is formally intractable.

Our overall-approach then is [3][5]:

(1) assemble the potential path-points into dense segments by using a fast MST algorithm (although the MST does not actually assure the densest connectivity, it usually provides a very good approximation to this condition). The input to this step is a binarized image; the output is a forest of (collection of disjoint) MST's.

(2) recover the longest segments – consistent with generic perceptual connectivity criterion –

that can be extracted from the forest of trees generated in step (1). (The list containing these segments is called RPATHS).

(3) repartition and semantically filter the collection of RPATHS to eliminate perceptual and semantic linking mistakes and irrelevant paths introduced or retained by the limited flexibility of the MST algorithm/representation and the generic parsing process.

(4) Use a very general linking technique and representation schema, capable of expressing arbitrary perceptual and semantic constraints, to imply a network of paths that is very likely to include the POP.

(5) Parse the network produced in (4) to extract a relatively small collection of prominent paths that includes the POP.

(6) Rank the paths extracted in (5), using an objective function based on the primary Gestalt criterion, and return the highest ranked path as the POP.

In the next section, we discuss some of the details of how the Semantic Delineation Subsystem (Figure 1) accomplishes steps 3 through 6.

## 3 The Semantic Delineation Subsystem

The Semantic Delineation Subsystem is composed of two major components; the Semantic Filter and the Semantic Linker. The Semantic Linker, in turn, has three main functional elements: (a) the SL-Segment-Linker, (b) the SL-Path-Generator, and (c) the POP-Generator.

### 3.1 The Semantic Filter (SF)

The purpose of the semantic filter is to extract, from a collection of perceptually salient paths, those sub-paths that are compatible with the constraints of some specified application or purpose (e.g., sub-paths that could be road segments in an aerial image).

This system component takes as its input a list of generic perceptually-salient paths (RPATHS) and produces, as its output, a list of path-segments (RPATHS-F). Each item (called a seg) in RPATHS-F, is a coherent sub-path of some path in RPATHS; the segs returned in RPATHS-F are open and non-self-intersecting, and any pair of segs are disjoint with the possible exception of a single intersection-point (as are the paths in RPATHS).

The SF processes each path in RPATHS independently. It first partitions the path into adjacent segs at it's salient points using the algorithm described in [4]. This partitioning step is necessary to recover components of the application relevant paths that were combined with other (incidental) adjacent paths in the original image. Each seg is evaluated for compatibility with the constraints of the intended application on an accept or reject basis. The accepted segs are appended to the output-list RPATHS-F. In addition, if two accepted segs were part of the same input (RPATHS) path, but are now separated in the sense that some portion of the input path between them was deleted by the filter, then an entry recording this fact is made on a link-list (see discussion of the Semantic Linker).

While the SF might have to be completely redesigned for each new application, we have found that the same set attributes (properly parameterized for the different applications) appears adequate for such diverse tasks as finding roads or rivers in aerial images, and for finding man-made objects (e.g., building edges) or natural objects (e.g., the skyline, tree-trunks) in ground-level images.

The attributes we currently evaluate (to be described in detail in a later version of this paper) are concerned with length, directionality, smoothness, and degree of randomness:

(1) Length. Very short segs are typically rejected as being "noise" or unimportant (they can be recovered later if necessary); very long segments are typically accepted since they are too important to discard without the further analysis to be performed later.

(2) Consistency of global direction based on a histogram of the directions between adjacent seg pixels obtained from a chain-coded representation of the seg.

(3) Smoothness. This property is measured in two ways. First, each seg is inherently smooth to some degree because its parent in RPATHS was partitioned into segments at salient (or high curvature) points. Thus, the length of the seg is an indirect measure of its smoothness (the longer the seg, the smoother it is). Second, we measure the seg's deviation from a best fitting circular-arc to look for a smoothness property that is especially important for some applications (e.g., finding man-made objects).

(4) Randomness. We have devised a weak measure of symmetry, or of repeated structure, in a path; this measure together with the evaluation of coherent length, consistent direction, and smoothness, provide a basis for judging whether a seg is a "purposeful" or an apparently random structure.

An example of the performance of a semantic filter we designed for delineating roads in aerial images is shown in figures 2d and 3d. Tables 1 and 2, in the section on experimental evaluation, presents quantitative results of the filtering operation in terms of a relevant set of semantic categories.

## 3.2 The Semantic Linker (SL)

The purpose of the semantic linker is to combine all the segs in the list RPATHS-F (produced by the Semantic Filter) into either a network of partitioned or unpartitioned paths, or to select and sequence a subset of the segs in RPATHS-F into a single POP; the problem of producing an unpartitioned network (generally, the more useful of the available types of output since a distinguished POP might not even exist) is a very simple sub-problem of producing a POP.

The SL has three components, (a) the SL-Segment-Linker, (b) the SL-Path-Generator, and (c) the POP-Generator.

### 3.2.1 The SL-Segment-Linker (SLSL)

The input to SLSL is RPATHS-F, and its output is the "link-pair-list." The SLSL examines every pair of segs in RPATHS-F and determines if they can be adjacent components of an extended path compatible with the constraints of the specified application. If so, it generates a "link-pair" entry which is appended to the "link-pair-list."

The SLSL typically uses three types of criteria to make a link decision for a pair of segs:

(1) The relative geometric positioning and separation of the segs. For example, in the case of road delineation, the criterion is typically a bound on the separation-distance between nominally corresponding endpoints (one on each seg). In the case of skyline delineation, the segs might be further constrained not to have any overlap in their horizontal (x) coordinates.

(2) Global attributes of the segs. For example, in the case of road delineation we might require that the spectral distribution, or image intensity, or mean width of the two candidate segs be identical to within some specified tolerance.

(3) Acceptance by the semantic filter. If the two candidate segs are linked as proposed and treated as a single seg, a sufficient condition for linking is that the combination is accepted by the semantic filter.

### 3.2.2 The SL-Path-Generator (SLPG)

The input to the SLPG is the "link-pair-list" produced by the SLSL and augmented by additional link-pairs supplied by the Semantic Filter; the output is either an unsegmented network (actually, a disjoint collection of such networks) or a pair of lists containing all possible maximal open-paths and loops implied by the link-pairs. The POP is assumed to be one of these (explicit) paths, and a simple test is proposed as a way of selecting it.

The function of the SLPG is purely syntactic/algorithmic – to expand the path information implicit in the augmented link-pair-list. The link-pairs are a compact encoding, actually generators, of the network or collection of paths to be produced by the SLPG.

We can easily partition a collection of link-pairs into disjoint subsets so that every pair of link-pairs referring to a common seg are in the same subset called a "link-pairs-association-set." The collection segs corresponding to such a subset is called a "seg-association-set." Each seg-association-set implies a disjoint network of paths (e.g., roads); networks consisting of a few short isolated paths can often be discarded as noise. The larger networks are typically returned as one of the major end-products of the system described here when used to find all the salient paths (e.g., roads or rivers) in an image. In this paper we are primarily concerned with a second type of output: explicitly extracting the single most salient path (the POP).

The SLPG operates as follows: The link-pairs are first partitioned into disjoint subsets (link-pairs-association-sets); these subsets are then processed independently to extract their implied paths using a collection of algorithms. For the purposes of this paper we describe the LP-Basic-Path-Extension-Algorithm (BEA) in some detail, but only indicate the basis for the remainder of the full extraction process (see Appendix).

A maximal-path through an LP-network is one that cannot be a proper continuous subsequence of some longer path; we will call the endpoints of a maximal-path terminal-nodes. A loop is a maximal-path that begins and ends on the same terminal-node. One requirement of the SLPG is to explicitly list all maximal-paths.

If the BEA is given a terminal-node as a seed, it will iteratively generate all the maximal-paths that have the given terminal-node as (at least) one of their endpoints. It is both fast and simple to find all the free endpoints (nodes of degree one) of an LP-network given its associated list of link-pairs; each such free endpoint (called an ept) is a terminal-node of one or more of the maximal-paths. In an LP-network without loops, we can generate all the maximal-paths using the set of epts as seeds. Each maximal-path will be found twice, but this redundancy does not cause any problems. The redundant paths can be avoided at considerable additional complexity in the BEA algorithm, but it is simpler to just detect and delete them should this be necessary.

If the network contains loops, except for some unusual situations, the above procedure will still return all the maximal paths (including the loops). Each loop could be generated many times (an upper bound is the product of the number of epts and "entry-points" to the given loop). If we wish to be assured that all the maximal-paths are found, and also reduce the redundant discovery of the same loop, then we can proceed as above (if there are any epts, otherwise, pick any node as the first seed and later discard the initial set of non-maximal-paths). All terminal-nodes which are not already members of the list of seeds are added to that list whenever they are found. When a loop is returned by the BEA, we have to modify all the link-pairs that point to segs that are components of the loop. We inactivate all link-pairs that point to two loop-segs, and replace each link-pair that points to exactly one loop-seg with two new link-pairs; in one case the original loop-seg-link-atom is replaced by a link-atom that will insert into a non-terminating path that originally included one or more loop-segs a dummy-seg identifying the loop; in the second case the original loop-seg-link-atom is replaced by a link-atom that identifies itself as a terminal-node associated with the given loop. In a sense, we collapse the loops in the original LP-network and create a modified loop-free network in which the BEA (algorithm) is assured to return all the maximal-paths. Those returned maximal-paths that contain dummy-segs are easily rectified.

In summary, there are some interesting theoretical issues that must be addressed in order to understand how to make the SLPG more efficient, but the algorithm we have described is computationally acceptable, and it returns all the maximal-paths as required to allow the SLPG to correctly perform its function.

### 3.2.3 The POP-Generator

The list of maximal-paths (both open-paths and loops) returned by SLSL is assumed to contain the POP. The segments comprising these paths have been previously filtered to assure compatibility with the semantic constraints of the specified problem domain, and are perceptually salient with respect to (at least some of) the Gestalt laws of perceptual organization. In the present algorithm, the POP-generator does not alter any of the maximal paths but simply selects the one that maximizes a combination of both path-density and path-length. Actually, the product of path-length and $(\textit{path-density})^2$ where path-density is measured by (number-of-path-points)/(path-length).

## 4 Experimental Evaluation

In addition to a significant amount of previous informal testing and evaluation (some parts of the Linear Delineation System were applied to well over 1,000 images of different types and with different delineation goals), we are now engaged in developing a formal evaluation methodology, especially in regard to road delineation.

In a typical road delineation problem (Figure 2) the Delineation System was invoked without any manual intervention or parameter tuning. We started with a 768X638 pixel image (489,984 points) that resulted in a binarized version (step 1) with 55,480 potential road points (Fig 2b). As a result of the generic delineation process (step 2), we extracted 340 segments (RPATHS) containing 21,255 points (Fig 2c). We defined six semantic categories of interest (Narrow Road, Wide Road, Proto Road, Ambiguous, Background, River) and manually classified the pixels along the paths into these six categories. If the labeling of a given Rpath was mixed, we counted the contiguous segments with the same label as being distinct – thus we judged that there really were 375 semantically distinct segments containing 21170 points comprising the 340 actual RPATHS with an associated count of 21,255 pixels. (Because of double counting of segment and path intersection-points, there is a small discrepancy in the number of points in the actual Rpaths and in the semantically labeled segments).

Table 1 and Fig 2d show the effectiveness of the Semantic Road Filter in retaining road points/segments while eliminating the unwanted background and river points/segments. Since the Road Filter was designed to retain narrow road segments, other structures (wide-roads, proto-roads, ambiguous) that could possibly be roads were considered to have a "don't-care" status in our evaluation.

A "window" (Figure 3) was manually selected and extracted from Figure 2 to test the POP-delineation algorithm. Here we started with a 475X149 pixel image (70775 points) that resulted in a binarized version (step 1) with 7214 potential road points (Fig 3c). The extracted set of 23 RPATHS contained 3696 points (Fig 3d). Table 2 and Fig 3e show the result of applying the Semantic Road Filter; it returned 37 segments containing 3117 points in RPATHS-F and 22 link-pairs (in *aux-link-pair-list*). The SL-Segment-Linker produced 19 additional link-pairs, and thus a total of 41 distinct link-pairs were supplied to the SL-Path-Generator; these were classified as consisting of 6 ept-pairs, 34 interior pairs, and 1 closed pair. The SLPG returned 43 open paths and 115 paths containing loops; a total of 158 maximal-paths. This set of paths contained redundant entries; actually, there were 8 distinct open-paths and 4 distinct closed-paths (loops). Each path was assigned a ranking using the the product of path-length and $(\textit{path-density})^2$ metric presented in the preceeding section. The POP-generator then selected the highest ranking path (it happend to be one of the closed-paths) as the POP (Fig 3f). This was the desired delineation.

## 5 Discussion

The work described in this paper is part of an on-going effort to fully automate the process of delineating perceptually and/or semantically meaningful line-like structures appearing in both aerial and ground-level images of scenes consisting mostly of natural features (e.g., trees, vegetation, drainage, and terrain) as well as some man made objects (especially roads). Our intent in preparing this paper was, in addition to its nominal subject matter, to describe relevant components of the system being assembled

| | RPATHS | | | RPATHS-F2 | | |
|---|---|---|---|---|---|---|
| Category | # points | % points | # paths | # points | % points | # paths |
| Narrow Road | 5843 | 28 | 45 | 5322 | 46 | 40 |
| Wide Road | 2102 | 10 | 12 | 2082 | 18 | 12 |
| Proto Road | 2941 | 14 | 62 | 1671 | 15 | 46 |
| Ambiguous | 1579 | 7 | 37 | 425 | 4 | 19 |
| Background | 8192 | 39 | 210 | 1720 | 15 | 87 |
| River | 513 | 2 | 9 | 294 | 3 | 5 |
| Total | 21170 | 100 | 375 | 11514 | 100 | 209 |

**Table 1:** Categorized Delineations for FT-HOOD1 image. Total pixels = 768 x 638 = 489984

| | W1-RPATHS | | | W1-RPATHS-F2 | | |
|---|---|---|---|---|---|---|
| Category | # points | % points | # paths | # points | % points | # paths |
| Narrow Road | 3060 | 85 | 8 | 2935 | 94 | 7 |
| Wide Road | 0 | 0 | 0 | 0 | 0 | 0 |
| Proto Road | 63 | 2 | 1 | 46 | 1 | 1 |
| Ambiguous | 146 | 4 | 3 | 46 | 1 | 3 |
| Background | 319 | 9 | 10 | 88 | 3 | 6 |
| River | 0 | 0 | 0 | 0 | 0 | 0 |
| Total | 3588 | 100 | 22 | 3115 | 100 | 17 |

**Table 2:** Categorized Delineations for FT-HOOD1-W1 image. Total pixels = 475 x 149 = 70775

for this purpose, illustrate and quantify some of its current performance, and discuss some aspects of the conceptual basis for its design.

The problem of finding the POP in (some designated portion of) an image is a basic requirement for effective man-machine communication about images, as well as a challenging problem whose solution is required to accomplish some of the more general delineation tasks. In this paper, we provide an approach to the solution of this problem, and an algorithm that is applicable to a limited class of scene domains. The algorithm has performed well on a small set of test cases but a significant amount of additional testing will be required before be can be sure of its utility and robustness.

An important contribution of this paper is the introduction of the LP-representation (link-pair/LP-network) and associated machinery as a generalization of the conventional graph. The LP-network provides a very powerful way of dealing with linear structures; it provides almost complete generality in specifying connectivity (more than is possible with a graph), it provides a very compact description of the (implied) connected structures, and it admits reasonable algorithms for the common (relatively simple) situations to be expected in images of real scenes. On the other hand, because there are specializations of the linking problem that are *NP-hard*, there are no generally efficient algorithms for this purpose.

There are many open problems and obvious extensions of the work discussed in this paper. However, one of the more interesting extensions would be to find a way to duplicate human performance in the following type of situation:

Consider an image composed of a sequence of 50 equal signs typed in a row (i.e., ======================= ...). Also assume there is a solid horizontal line positioned just below the equal signs that has the same horizontal extent. If we assume that there are four links possible between each pair of successive equal signs (two straight links and two cross-over links; we ignore the vertical links which lead to short closed paths), then there are on the order of $2^{50}$ paths that potentially would have to be generated before we could decide – by applying some objective function to an explicit descriptions of the competing paths – if some path through the equal signs, or the solid line, or neither, was the POP. Obviously, the human doesn't do this; he picks the solid line almost immediately; how is he able to avoid the combinatorial explosion??

One of our main points in this paper, and the basis of our approach, was that much of the assembly of the ultimately to be selected POP had to take place in the generic perception phase which sacrifices flexibility and generality for simplicity and speed. The Semantic Linker is computationally limited and can't be handed a problem with too many choices. Thus, the overall delineation system must include mechanisms that enforce complexity constraints on the output of each of the its subsystems – this type of control could be accomplished by iteratively adjusting algorithm parameters. It thus appears that a vision system must be fully cognizant of its computational limitations if it is to operate effectively. Understanding how to accomplish this type of control is one of our more immediate goals.

# A Appendix

## A.1 Definitions

EPT: a "free endpoint" designates the end of a seg which is not referenced by any currently active link-pair; e.g., a path containing an ept cannot be further extended at that end.

LINK-ATOM: a list of two items, the first is an index number into RPATHS-F; i.e., it points to a seg in RPATHS-F. The second item is a logical variable (T or NIL) which specifies whether the seg is to be used as stored (NIL) or reversed (T).

LINK-LIST: a list of two or more link-atoms. It specifies how to assemble a path from a subset of the segs stored in RPATHS-F.

LINK-PAIR: a list of two link-atoms. It specifies a path consisting of the concatenation of the two segs in the order listed, with the points in each seg taken as stored in RPATHS-F, or reversed, as specified by the logical variables.

LINK-PAIR-LIST: nominally, the list of link-pairs produced by the SL-Segment-Linker and the Semantic Filter.

LOOP: a subsequence of a path that begins and ends with an identical link atom, or, a subsequence of a path that begins and ends with a link atoms pointing to the same seg, but having reversed directions.

LP-NETWORK: the collection of paths implied by a collection of link pairs.

CONNECTED-LP-NETWORK (CLPN): a collection of link-pairs can be partitioned into disjoint subsets so that every pair of link-pairs referring to a common seg are in the same subset called a "LINK-PAIRS-ASSOCIATION-SET." The collection segs corresponding to such a subset is called a "SEG-ASSOCIATION-SET." Each seg-association-set implies a disjoint network of paths called a CLPN.

PATH: a concatenation segs (segments) as specified by a link-list. No seg can appear more than once – with the exception of the seg specified by the head link-atom in the case of a loop or semi-loop. The HEAD of the path is intended to refer to the end at which the path is being extended; the TAIL of the path is intended to refer to the end of the path containing the seed link-atom, and we arbitrarily assume that the path is, or was, constructed by a sequential accumulation of segs starting at the tail-end. For most purposes, we further restrict this definition to prohibit the path from visiting a vertex more than once.

OPEN-PATH: a path that does not contain a (complete) loop.

MAXIMAL-PATH: A maximal-path through an LP-network is one that cannot be a proper continuous subsequence of some longer path; we will call the endpoints of a maximal-path TERMINAL-NODES. A LOOP is a maximal-path that begins and ends on the same terminal-node.

## A.2 Some Attributes of an LP-network

1. If an LP-network has no loops, then the terminal-nodes of every maximal-path are epts; i.e., every fully extended path will begin and end at an endpoint of a seg which is not connected (by an active link-pair) to any other seg. If the LP-network does contain loops, then the "entry-points" to the loops will also be terminal-nodes of maximal-paths.

2. There might not be any single path connecting two given epts, even in a connected-component of the network. For example, consider a network consisting of three segs connected as a Y. If the two upper arms of the Y both connect to the lower vertical stem, but not to each other, then there is no direct path linking the two free ends of the upper segs of the Y.

3. An LP-network can always be converted into a conventional graph by adding additional link-pairs so as to make every vertex "fully-connected."

## A.3 Basic Algorithms

Key algorithms include:

- lp-basic-path-extension-algorithm
- partition-lp-network-into-connected-subnetworks
- identify-ept-lp-algorithm

### A.3.1 LP-BASIC-PATH-EXTENSION-ALGORITHM

For a given subset of link-pairs (say set q), a link-atom (say s1) is selected from one of the link-pairs in q as a seed, and a collection of paths are iteratively constructed from this seed by successively scanning all the link-pairs (in q) for additional segs to append to the set of paths still being extended. We note that a path is represented by a link-list (a list of link-atoms); all the partial paths we are currently extending have one endpoint defined by the initial seed s1. Consider a particular path (say p7) which has as its current terminal link-atom the list (seg24 t). If some link-pair in q has the form ((seg24 t) (segX t/nil)), then p7 can be extended by appending link-atom (segX t/nil) to its current link-list. There may be more than one link-pair in q capable of extending the current version of p7. Further, if some link-pair in q has the form ((segZ nil/t) (seg24 nil)), this link-pair representing the concatenation of segs segZ and seg24, is equivalent (in that the two segs are joined in the same way) to the link-pair represented by ((seg24 t) (segZ t/nil)), and thus p7 could be extended by appending link-atom (segZ t/nil).

At each iteration we start with a list of partial paths, and for each such path there are three possibilities: (a) there are no further extensions, in which case the path is placed on the output open-path-list; (b) there is an extension, but the extending seg (or the unattached vertex of the seg) already appears in the partial path, in which case the extended path is placed on the output closed-path-list; (c) there are one or more (single seg) extensions with new segs, in this case, all such extensions are placed on a new partial-path-list and the above process is repeated. The process terminates when the partial-path-list has no entries at the start of a new iteration.

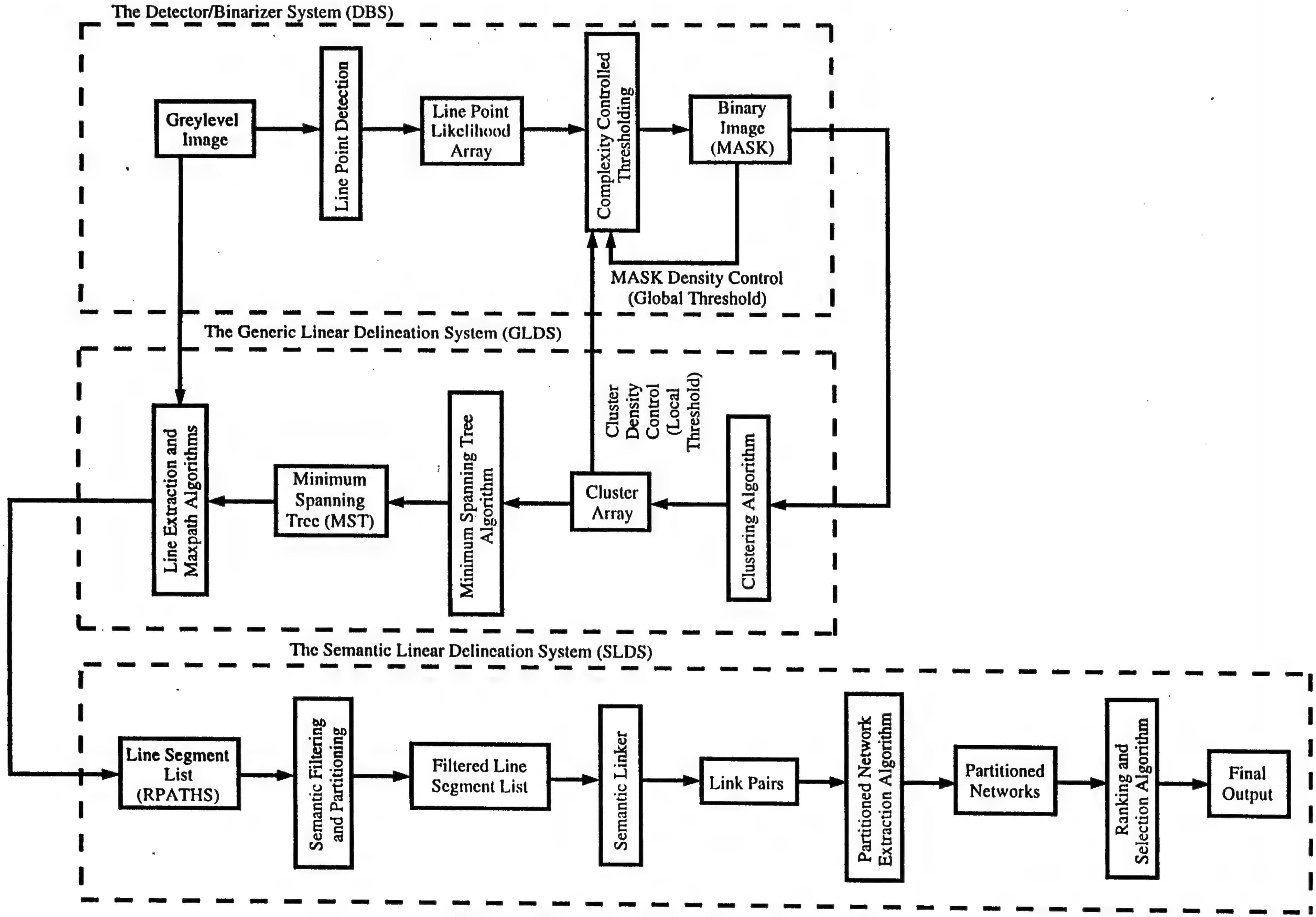
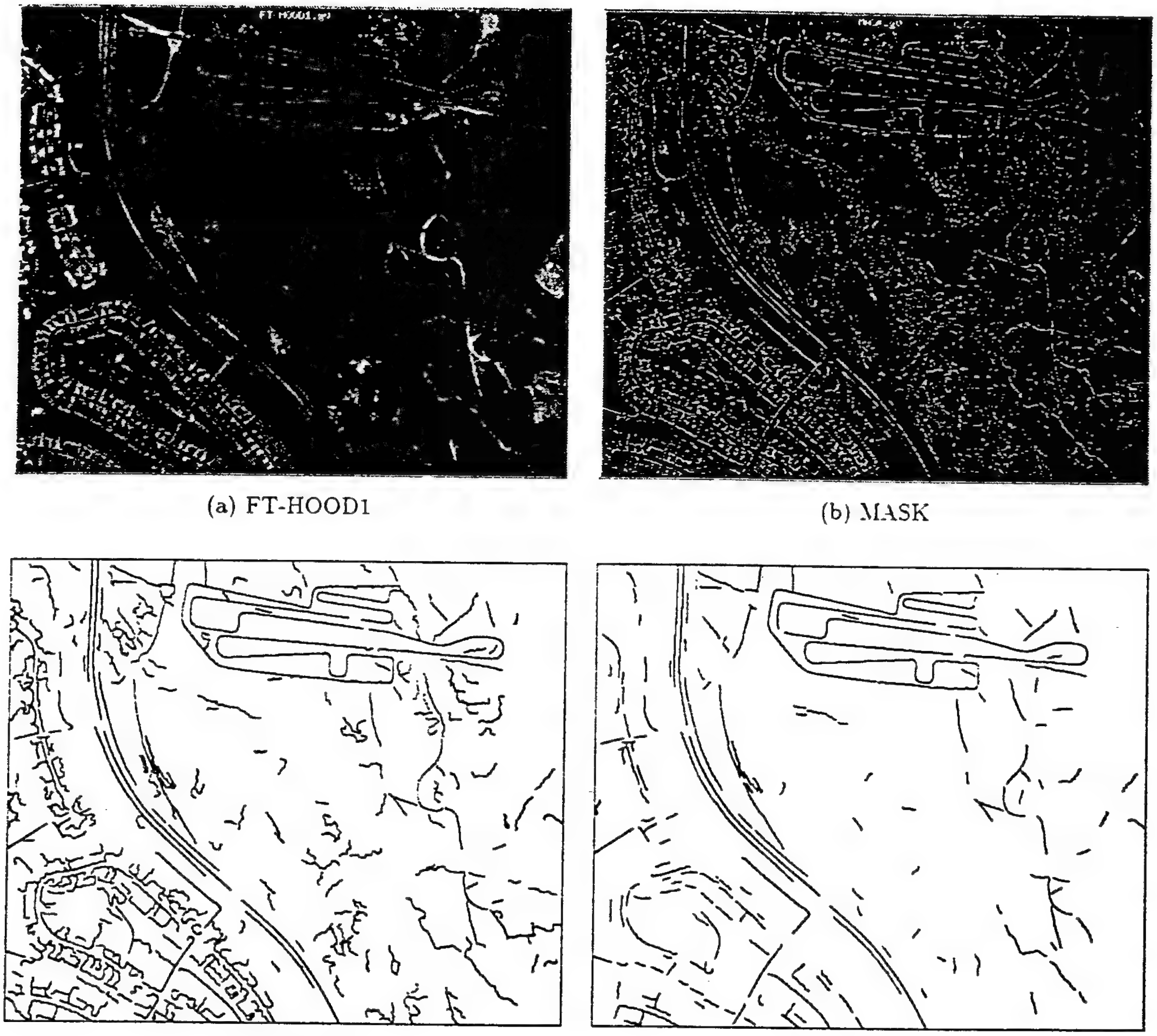

The Detector/Binarizer System (DBS)
Greylevel Image
Line Point Detection
Line Point Likelihood Array
Complexity Controlled Thresholding
Binary Image (MASK)
MASK Density Control (Global Threshold)
The Generic Linear Delineation System (GLDS)
Line Extraction and Maxpath Algorithms
Minimum Spanning Tree (MST)
Minimum Spanning Tree Algorithm
Cluster Array
Cluster Density Control (Local Threshold)
Clustering Algorithm
The Semantic Linear Delineation System (SLDS)
Line Segment List (RPATHS)
Semantic Filtering and Partitioning
Filtered Line Segment List
Semantic Linker
Link Pairs
Partitioned Network Extraction Algorithm
Partitioned Networks
Ranking and Selection Algorithm
Final Output

Figure 1 The Linear Delineation System (LDS)



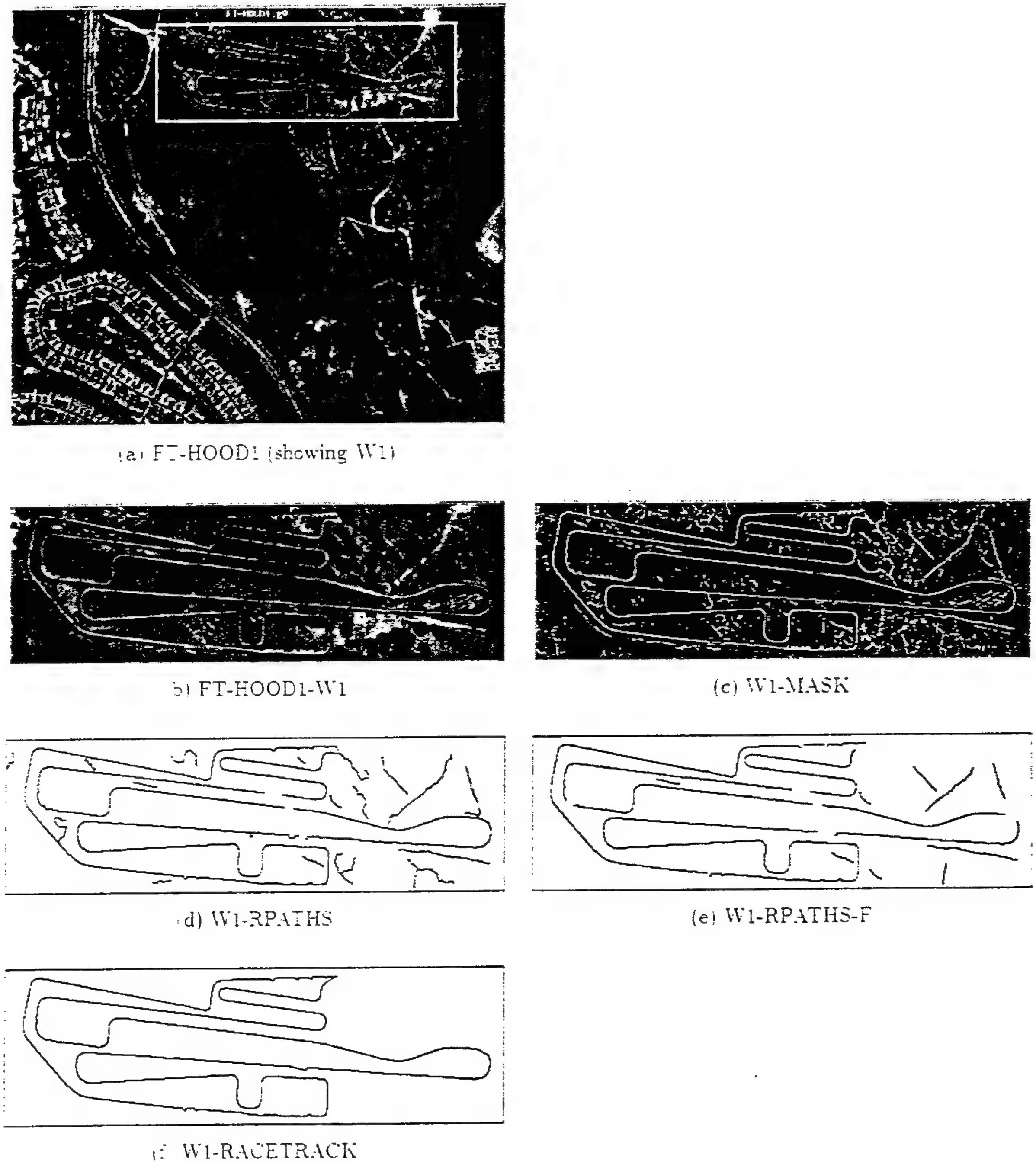
(a) FT-HOOD1

(b) MASK

(c) RPATHS

(d) RPATHS-F

Figure 2: Extracting Roads from an Aerial Image



(a) FT-HOOD1 (showing W1)

(b) FT-HOOD1-W1

(c) W1-MASK

(d) W1-RPATHS

(e) W1-RPATHS-F

(f) W1-RACETRACK

Figure 3: Extracting a POP from an Aerial Image

## References

[1] M.A. Fischler and O. Firschein, "Intelligence: The Eye, the Brain, and the Computer," (331 pgs.), Addison Wesley, 1987.

[2] M.A. Fischler, J.M. Tenenbaum, and H.C. Wolf, "Detection of roads and linear structures in low-resolution aerial imagery using a multisource knowledge integration technique," Computer Graphics and Image Processing, vol 15(3), March 1981, pp 201-223; also, Readings in Computer Vision (M.A. Fischler and O. firschein, eds.), Morgan Kaufmann, pp 740-752, 1987.

[3] M.A. Fischler and H.C. Wolf, "Linear Delineation," Proceedings IEEE CVPR-83, June 1983, pp 351-356; also, Readings in Computer Vision (M.A. Fischler and O. firschein, eds.), Morgan Kaufmann, pp 204-209, 1987.

[4] M.A. Fischler and H.C. Wolf, "Locating perceptually salient points on planar curves," IEEE PAMI vol. 16(2):113-129, Feb. 1994.

[5] M.A. Fischler, "The Perception of Linear Structure: A Generic Linker," Proc. ARPA Image Understanding Workshop. Monterey, Calif, Nov, 1994.

[6] A. Gibbons, "Algorithmic Graph Theory," Cambridge Univ. Press, 1985.

[7] E.L. Lawler, et. al., "The Traveling Salesman Problem," Wiley-Interscience, 1985.

# APPENDIX B:

## “Robotic Vision: Sketching Natural Scenes”

M.A. Fischler

Proc. ARPA Image Understanding Workshop,
Palm Springs, CA February 1996

# Robotic Vision: Sketching Natural Scenes

Martin A. Fischler
SRI International
333 Ravenswood Avenue, Menlo Park, CA 94025, USA
(fischler@ai.sri.com)

## Abstract

The goal of the work described here is to advance the state of the art in scene interpretation for autonomous systems that operate in natural terrain. The primary contribution of this paper is the presentation of a few key ideas and corresponding computer algorithms that can create a primitive labeled scene sketch (from images of an outdoor scene) that makes explicit the qualitative geometry and identifies the major visible natural objects and terrain features. A central problem is the necessity to replace reliance on generally unavailable explicit shape with more general ways of recognizing and describing natural objects. Our approach is to first select, or define, a small set of primitive (but pervasive) features that can be reliably extracted from most images of natural scenes. This paper is primarily concerned with the extraction and immediate utility (e.g., for reactive robotic vision) of these features currently consisting of: color, texture, shadows, depth, surface orientation, and linear structures

## 1 INTRODUCTION

The ultimate goal of the work described here is to advance the state of the art in scene interpretation for autonomous systems that operate in natural terrain. Such systems are currently unable to model (understand) their surroundings much beyond the direct use of information available from local geometric shape recovery methods (e.g., stereo, sonar). There is currently no reliable technology that can be used to recognize and semantically label natural objects and terrain features appearing in images of outdoor scenes.

The problem of automatically recognizing objects appearing in images of the outdoor world has proven to be extremely difficult, at least in part, because of the lack of explicit shape models for such objects. Most computer-based recognition techniques rely on explicit knowledge of shape, but rocks, trees, and other natural objects cannot be successfully described in this way; even such generic man-made objects as roads, bridges, and buildings are more likely to satisfy functional constraints rather than being exemplars of some geometric blueprint. It is necessary to replace explicit shape with a more general way of describing natural objects and complex man-made structures.

The primary contribution of this paper is the presentation of a few key ideas and corresponding computer algorithms that can create a primitive labeled scene sketch (from images of an outdoor scene) that makes explicit the qualitative geometry and identifies the major visible natural objects and terrain features. As noted above, a central problem is the necessity to replace reliance on generally unavailable explicit shape with more general ways of recognizing and describing natural objects. Our approach is to first select, or define, a small set of primitive (but pervasive) features that can be reliably extracted from most images of natural scenes. This paper is primarily concerned with the extraction and immediate utility (e.g., for reactive robotic vision) of these features currently consisting of: color, texture, shadows, depth, surface orientation, and linear structures. In previous publications, we described how such primitive features can be combined to identify clear instances of the natural objects of interest using a "production rule" type paradigm [Strat91], and then, using these recognized objects as exemplars, we can invoke (say) a nearest-neighbor statistical classifier to label other, possibly less obvious, instances of the objects we are looking for. Objects of interest (the semantic vocabulary) include rocks, trees, brush, grass, water, snow, ground, sky, ridgelines, holes/ditches, roads, paths, fences, poles, cliffs, ground-plane, and shadows.

# 2 KEY IDEAS

## 2.1 Reduced Representations

The information we wish to extract from an image must be assumed to be encoded in the color/intensity arrays that define the image. The problem is that each image represents a collection of unique instances of the set of the much more general categorical labels we wish to assign to the image and its components. The value of categorical labels is that they provide a compact set of keys for indexing into a knowledge data base, or a decision table (say for reactive behavior). Thus, the problem of image interpretation can be viewed as computing a Reduced (generalized) Representation of the given image. We exploit this view in two ways: (a) we construct Reduced Representations of the original modalities (color and intensity) as intermediate abstractions which retain sufficient information content to retrieve the ultimately desired labels; and (b) the labels for the intermediate abstractions are a subset of the original colors/intensities themselves, and the semantic categories they imply are those entities with the given colors/intensities as attributes. *We require that the Reduced Representations we construct be suitable directly viewable replacements for the original image,* and we accept the fact that semantically distinct objects can be identically encoded (as can be the case even in the original image) and that objects in the same semantic category can be differently encoded (as can also be the case in the original image).

The line-sketch, the texture-overlay (texture-sketch), and the thresholded-image (shadow-sketch) are representational forms that could be Reduced Representations of a graylevel image (according to our above definition) if the appropriate semantic rules and constraints are employed in their construction and interpretation; we provide such rules and introduce the "Color-Sketch" as a Reduced Representation for color images.

The requirement that the Reduced Representation be a directly viewable replacement for the original image distinguishes it from the more conventional "feature array." It assures us that in spite of the greatly simplified description provided by the Reduced Representation (six orders of magnitude, from 24 bit original color to 4 bits of false-color for the color-sketch) we have retained the intrinsic relationships and constraints that the human visual system uses to recognize a picture as depicting a natural scene. The person viewing the color-sketch does not need to know that we have assigned specific semantic categories to the individual colors; the sketch can still be properly interpreted. Without the *directly viewable* requirement, the conventional feature array is an arbitrary symbolic encoding that has no intrinsic content; it can't be interpreted without knowing what the pixel values are supposed to represent. It is very difficult to determine (either by a external observer or an internal critic) when the encoding algorithm has made a mistake.

## 2.2 Regularities Rather Than Invariants

The value of the Reduced Representation, constrained to be a directly viewable replacement for the original image (at least for some particular purpose), is that it must capture and reflect some important physical regularity – not necessarily a physical law that can't be violated, but correct often enough to give the right gestalt to a human observer. As we show below, such regularities allow us to (a) provide a raised-object overly from intensity thresholding (the shadow-sketch); (b) to provide a surface-orientation overly from texture (the texture-sketch; (c) to provide a delineation of the skyline-ridgelines from line and edge operations (the line-sketch); and a false-color-image encoding of semantic labels by "collapsing" the true colors in a color image (the color-sketch). These four sketches (and a few others not yet implemented) comprise what we call the "Scene Sketch." The (viewer-centered) Scene Sketch can be used directly to control the immediate behavior of a robotic device, or it can provide the primitive information needed by a CONDOR-type system [Strat91] to derive an objective high-level description needed for longer-range planning and decision making.

# 3 The Scene Sketch

The Scene Sketch is currently a composition of the four implemented components described below (color-sketch, shadow-sketch, line-sketch, texture-sketch). Additional components could include a depth/terrain-profile sketch and a polarization-sketch.

## 3.1 The Color-Sketch and "Recoloring" Algorithms

A very useful semantic description of a scene can be based on the categories: sky/clouds/snow, live-vegetation, earth/ground/rocks, water/rocks, shadows/unknown. This list, which can obviously be refined and expanded, appears to be within the competence of pixel-level color-image processing given that we allow the somewhat intermixed categories; e.g., a volcanic rock can look quite dark or "ground-like," while under suitable lighting and view conditions, (some portion of) a rock composed of granite will reflect light just like the surface of a body of water – and is indistinguishable from water just using local color.

### 3.1.1 Regularities

It is almost axiomatic that, at least to the human visual system (usually) the sky is blue, vegetation green, the earth gray/red/brown, water blue/green, etc.; what actually appears in an image is somewhat different. We have devised a recoloring algorithm that exploits color and intensity regularities present in most normal outdoor scenes. In particular, the relative blue content and brightness of a pixel both vary according to the high-to-low scale: cloud/snow/sky; water/rock; ground; live-vegetation; shadows.

Recoloring is not an attempt at color constancy or color partitioning!! The different semantic categories are each composed of many different shades of the nominal color we assign, and the Color-Sketch is a pixel-level description – we do not explicitly delineate closed regions of constant color/material type.

The semantic considerations underlying the rules invoked by the automatic recoloring algorithms are discussed below.

- Sky

  The blue appearance of the sky in a color image is the result of human physiology and psychology, as well as physics – the red or green component at a blue appearing image-pixel can often be more intense (but only by a small percentage) than the blue component. The unclouded sky, with the exception of the sun itself, can usually be assumed to be the brightest object in the image (illuminated clouds and snow can be much brighter than blue sky, but clouds and snow are included in our "sky" semantic category. The sun radiates most brightly in the orange-yellow wavelengths, and the wavelength selective scattering of the air molecules disperses the blue light component of the suns ray's much more strongly than the longer wavelengths (Rayleigh's law: scattering is inversely proportional to the fourth power of the wavelength). If we look at the clear sky directly overhead (top of the image), with the sun off to the side, the scattered blue light dominates. As we shift our gaze in the direction of the sun (toward one of the sides of the image), the relative strength of the longer wavelengths increases; for more complex reasons [Minnaert], as we lower our gaze toward a distant horizon (a longer path in the atmosphere than looking straight up) the various selective factors tend to equalize and the light appears white (unsaturated rather than colored). There are may complicating factors (e.g., large dust or water particles in the air, reflections from clouds, snow, or the earths surface, polarization effects) that a sophisticated system might want to consider, but the nominal conditions discussed above seem to be an adequate base for constructing the sky coloring rules used by our color sketching algorithm.

- Water

  The appearance of water depends critically on whether the observer/camera is seeing a reflection or viewing the scene content below the water's surface. For a ground-based observer in open terrain, it is reasonable to assume that much of the light coming from a body of water is reflected sun or skylight. Thus, we expect water to be bright with a significant blue spectral content. To distinguish sky from water, we can take advantage of the fact that both the unclouded sky and clear water are highly polarized, but in different directions. Further, we would expect that water would have a larger green spectral component than direct skylight.

- Rocks

  Rocks and soils are aggregates of minerals. A mineral has a regular atomic arrangement and it has a narrow range of chemical and physical properties. Rocks are generally classified as either sedimentary (e.g., limestone, shale), igneous (e.g., basalt, plutonic), or metamorphic (e.g., slate, schist). Granite, the most common type of rock appearing in our small image collection, is a coarse-grained rock composed chiefly of feldspar and quartz – its origin could be either igneous or metamorphic [Gilluly59]. For a simple visual system, rocks pose quite a challenge – just recognizing their presence, let alone their type. It appears that the most effective way of detecting rocks is either by the shadows they cast, or by the fact that they (at least the granites) are good reflectors of the suns light. In our experiments, we found that there was very little difference in the spectral composition of light reflected from granite or from water, so we lumped them into a single semantic category (i.e., they receive the same color in our false-color reduced representation image). It is relatively easy to separate them (rocks and water) in a subsequent analysis step; e.g., water generally does not cast a shadow and is not a raised object.

- Live-Vegetation

  Almost all live vegetation possesses chlorophyll; chlorophyll has a sharp reflectance peak (relatively low absorption) for green light (510-580 nm). The various shades of green we see in different types of vegetation is due to additional reflectors (pigments) which largely affect the longer wavelengths. When the chlorophyll dies, we see the effects of these other reflectors: yellows, reds, but not blue. Almost all green leaves are uniformly good absorbers of wavelengths shorter than 500 nm [Lythgoe79]. Thus, a relatively low blue spectral component (especially compared to green) at an image location is a strong cue for the presence of vegetation. Live-vegetation absorbs most of the light that strikes it; e.g., timberland reflects 3 percent and open grassland 6 percent, compared to concrete at 36 percent and snow at 80 percent [Remote Sensing70]. Thus a second cue for the presence of live-vegetation is low intensity; the fact that vegetation is frequently self-shadowing further enhances the probability that a dark region contains vegetation.

- Ground

  Ground is currently a "catch-all" category that includes soil, low-lying rock, dead vegetative ground cover, etc. In a sense, if a pixel is not identified as belonging to one of our other semantic categories, and it falls between water/rock and live-vegetation in both the blue spectral intensity and absolute intensity ranges, we currently label it as ground.

### 3.1.2 Algorithm

Appendix A presents the completely automatic recoloring algorithm we have devised, and Figure 1 shows an example of its performance. We have also constructed a "nearest neighbor" learning-type pixel-level classifier that works very well, and can recognize a much larger set of semantic categories than the automatic recoloring algorithm – but it is dependent on an external source of labeled training samples, and is likely to be sensitive to the peculiarities of the specific scene/images from which the training samples were derived.

## 3.2 The Shadow-Sketch

Shadows appear to be critical to the human visual system (HVS), at a primitive level, for the purposes of perceptual organization and geometric scene understanding. It is easy to demonstrate that the HVS becomes confused when the shadows in an image are disguised by making them brighter (rather than darker) than the surfaces they project to [Cavanagh89].

As long as there is any light in the sky, we can usually discern (at least) the local scene geometry. In art there is a style called "Chiaroscuro" which uses just two graylevels (uniformly black marks on a white background) to depict scenes, and a "four value" style which appears quite adequate to produce very clear renderings of natural outdoor scenes [Johnson90]. Figure 2 shows some examples of the adequacy of properly thresholded (two-level) images.

A primary use of shadows is as an indicator of raised objects – the ground surface does not cast a shadow nor is it typically self occluding. Shadows, in an outdoor scene, can be defined as being present at those locations that are not directly illuminated by the sun. Because of illumination by reflected light, by diffuse light, and the presence of darkly colored objects, it is not always obvious by direct inspection where shadows actually occur. The critical observation (regularity) we exploit is that almost any grayscale image can be reduced to a one bit depiction of a scene, by almost any reasonably low threshold, and still alow a human observer to correctly interpret the qualitative scene geometry. A smaller threshold will cause some potential obstacles to be missed, a higher threshold will mark some darker, but un-raised regions as obstacles.

The algorithm we employ to set the threshold for computing the Shadow-Sketch first scans the image looking for pixels with a large local intensiy gradient (in the top 20 percent of the gradient values found in the image); it assumes that these pixels lie on the boundary of a shadow and identifies the darkest pixel in a 5X5 square about boundary pixel as a shadow-pixel. The algorithm then histograms the intensity values of these detected shadow-pixels and sets the shadow-threshold at the mode of the collection if the mode falls within one standard deviation of the mean; otherwise, it sets the shadow-threshold at one standard deviation above the mean.

## 3.3 The Line-Sketch

The Line-Sketch is the most obvious and well known form of Reduced (visual) Representation. Human-produced artistic renderings provide an existence proof that a Line-Sketch can usually be constructed as a directly viewable replacement for an image of a scene. It appears that good line sketch makes explicit physical and geometric discontinuities in material-type, illumination, depth (i.e., occlusion boundaries), and surface orientation. In spite of the seeming simplicity of the task, fully automated methods for constructing a Line-Sketch do not (yet) exist.

Our work on constructing a Line-Sketch as a component of the Scene Sketch is currently limited to finding sky-lines (see Figure 2) and ridgelines. We have a simple but effective algorithm for finding individual skyline points given the availability of the Color-Sketch: after first removing small isolated clusters of "sky-points," we mark the location of the lowest (smallest y-coordinate) sky-point in each column of the image array. While explicit linking is not necessary for the Line-Sketch, it is needed to properly extend the skyline through occlusions (which, in turn, is necessary for a simple technique to detect raised vegetation). We invoke the linking techniques presented in [Fischler94] for this purpose. Work on extending the Line-Sketch to include ridgelines is nearing completion.

### 3.4 The Texture-Sketch

The shading and texture variations in a graylevel image frequently allow a human observer to correctly infer the orientation of the visible scene surfaces. There currently is no computational theory that can explain or duplicate this human ability in the case of natural outdoor scenes. Further, there is no assurance that a Reduced Representation can be strictly based on the depiction of texture or shading information. Nevertheless, because of its utility (should we be successful), we are attempting to construct a Reduced Representation, based on texture, which exploits two regularities:

- raised (relatively isolated) objects have significant vertical edge content
- because of "foreshortening," most of the projectively imaged edges on a relatively horizontal plane in a typical outdoor scene will appear to be horizontal in the image

Rather than explicitly extracting edges and attempting to analyze them, we are currently attempting to identify isolated raised objects and horizontal (support) surfaces by finding coherent regions where the local intensity gradient is predominantly vertical or horizontal (see Figure 3).

## 4 ASSUMPTIONS, VISUAL CONTEXTS, AND EXPERIMENTAL RESULTS

This paper primarily focuses on a visual environment consisting of open "rolling" terrain with (possibly) water bodies, distant mountains, scattered trees or clumps of trees, and brush – there can also be nearby cliffs and ravines. Other natural environments (which we do not address) could include operating in deep forests, under-water, or surrounded by extreme relief (as in mountain climbing).

It is nominally assumed that the camera is approximately 4-6 feet above the ground with its principal axis horizontal (i.e., normal to the direction of gravity).

While not because of any actual constraints, most of the pictures used in this study were taken while the sun was illuminating the scene from a point at least 30 degrees above the horizon, and some portion of the sky (possibly clouded) was visible in the image.

Most of this work is based on analyzing single color images with generally unknown camera parameters, from a database of approximately 30 images (90 color-component images). The pictures come from a variety of sources, but primarily the foothills near Stanford University, California deserts, and the California Sierra's at high elevations (above 8,000 feet).

All the experiments performed to-date were informal, but since the primary criterion for success is that the computed Reduced Representation be perceived by a human observer as being qualitatively equivalent to the original image, it will be difficult to devise a useful scoring procedure that is more precise than success/failure. We intend to tabulate this type of result on a larger data-set once the current collection of algorithms is deemed to be stable.

The algorithms that produce the Color-Sketch, the Shadow-Sketch, and the Line-Sketch, generally seem to provide very good results although they can be improved. The Color-Sketch has too small a vocabulary for some scenes; it especially needs a separate label for (distant/haze-shrouded) mountains – at present, they are often included in the sky region. The line sketch is far from complete since, at present, it only consists of the skyline; however, a ridgeline detector is nearing completion. The texture sketch we currently compute is more problematic, it often provides good results that are valuable and not easily obtained from other sources, but it not as reliable as we would desire (its performance degrades with distance), and the "image-overly" it provides cannot generally be recognized as a replacement for the original image: thus, it is not yet a Reduced Representation, but rather a more conventional feature array.

## 5 DISCUSSION

To the naive eye, usually, the sky is blue, vegetation green, the earth gray/red/brown, water blue/green, etc. Is it possible to take a real color image, and on a local (or even pixel level) basis, produce a "false" color image with a few colors (say 4-16), each color corresponding to a specified semantic category, and the false-color image itself a recognizable replacement for the original – not only with respect to semantic labels, but also allows recovery of gross terrain geometry?? If such recoloring is indeed possible, as our initial experiments seem to imply, the implications are quite profound. Such an easily derived explicit representation (the Color-Sketch) could provide a way for a simple organism (animal or animate – without conventional language machinery, higher level reasoning, or sophisticated mathematical manipulation), to base immediate (visually-guided) behavior on semantic considerations.

We have extended the above idea, that of the Reduced Representation, to extract iconic overlays identifying raised objects (from the graylevel image, identifying horizontal and vertical structures (from the texture overlay), and the skyline, as a second order simplification, from the color sketch.

In attempting to design a vision system for a robotic device (even a vision system limited to supporting the task of outdoor navigation) and encountering a host of refractory problems, one can't help wondering how simple biological organisms can, seemingly, perform this task so well. Are we missing something very obvious? While our nominal concern is ultimately to support a full range of interactions of the robot with its environment, a more achievable initial objective is to consider only those aspects of visual interpretation required for local navigation. The semantic vocabulary could be as simple as go/no-go directions open to the robot. It is more important to recognize such functionally meaningful image-point-attribute distinctions as solid/deformable, flat/raised, close/distant, than specifically recognizing that something is a tree rather than a rock. Nearby objects should be given more attention (with respect to positional accuracy and semantic resolution) than distant ones which can be dealt with again at a latter time if necessary. A subjective (viewer-centered) model, e.g. an iconic overlay of the image, that can be used for reactive behavior (as noted above) turns out to be relatively easy to derive (a major point of this paper) as compared to an objective model, e.g., a symbolic labeling of the partitioned scene, that is required for long range planning. To the extent that the sensing modalities are available, e.g., stereo, motion, color, and polarization, they can pay very high dividends in the simplification of the interpretation task over what can be accomplished with single graylevel images.

The Scene Sketch has direct utility for reactive robotic navigation since its overlays of the scene allows the robot to quickly determine the likely presence of raised objects, flat navigable areas, and surface material type in any view direction. This information is available in qualitative form even without the availability of explicit depth overlays (say, from stereo) or the need for explicit partitioning. A significant number of pixels with the same semantic or geometric label in a particular view direction tells the robot what it is likely to encounter if it moves in that direction. The vertical position (y-coordinate) of the first (smallest y-coordinate) pixel in a coherent sequence of identically labeled pixels provides an estimate of the distance to the corresponding object/region.

Since the Scene-Sketch is qualitative, and its vocabulary is limited, its appropriate use beyond reactive navigation is as input to higher level analysis processes. For example (see Figures 1 and 2), since any non-sky pixel in the color sketch located above the skyline (in the line-sketch) can be assumed to be a pole or raised vegetation (a tree or a large bush), we can easily extend the semantic vocabulary of the primitive scene sketch to include these additional objects and detect them with relatively simple algorithmic techniques. We can also invoke simple rules to check physical consistency; e.g., a pixel labeled water cannot (correctly) lie vertically above a pixel labeled sky.

## 6 ACKNOWLEDGEMENTS

The research reported here was funded by the Advanced Research Projects Agency and monitored by the U.S. Army Topographic Engineering Center under contract DACA-76-92-C-0008. The author also acknowledges the many contributions made by Helen Wolf to the work presented in this paper.

## 7 REFERENCES

1. P. Cavanagh and Y.G. Leclerc, "Shape From Shadows," Experimental Psychology: Human Perception and Performance 15(1):3-27, 1989.
2. M.A. Fischler, "The Perception of Linear Structure: A Generic Linker," Proc. ARPA Image Understanding Workshop, Monterey, Calif, Nov, 1994.
3. Gilluly, Waters, and Woodford, "Principles of Geology (2nd ed.)," W.H. Freeman and Co., 1959.
4. C. Johnson, "The Sierra Club Guide to Sketching in Nature," Sierra Club Books, 1990.
5. J.N. Lythgoe, "The Ecology of Vision," Clarendon Press, Oxford, 1979.
6. M. Minnaert, "Light and Color," Dover, 1954.
7. Agricultural Board, National Research Council, "Remote Sensing," National Academy of Sciences, 1970.
8. H. Rossotti, "Color," Princeton Univ Press, 1985.
9. Strat, Thomas M., and Martin A. Fischler, "The Role of Context in Computer Vision," Proceedings of the IEEE Workshop on Computer Vision, Cambridge, MA, June 1995.
10. Strat, T.M. and M.A. Fischler, "Context-Based Vision: Recognizing Objects using both 2D and 3D Imagery," IEEE PAMI 13(10):1050-1065, Oct. 1991.

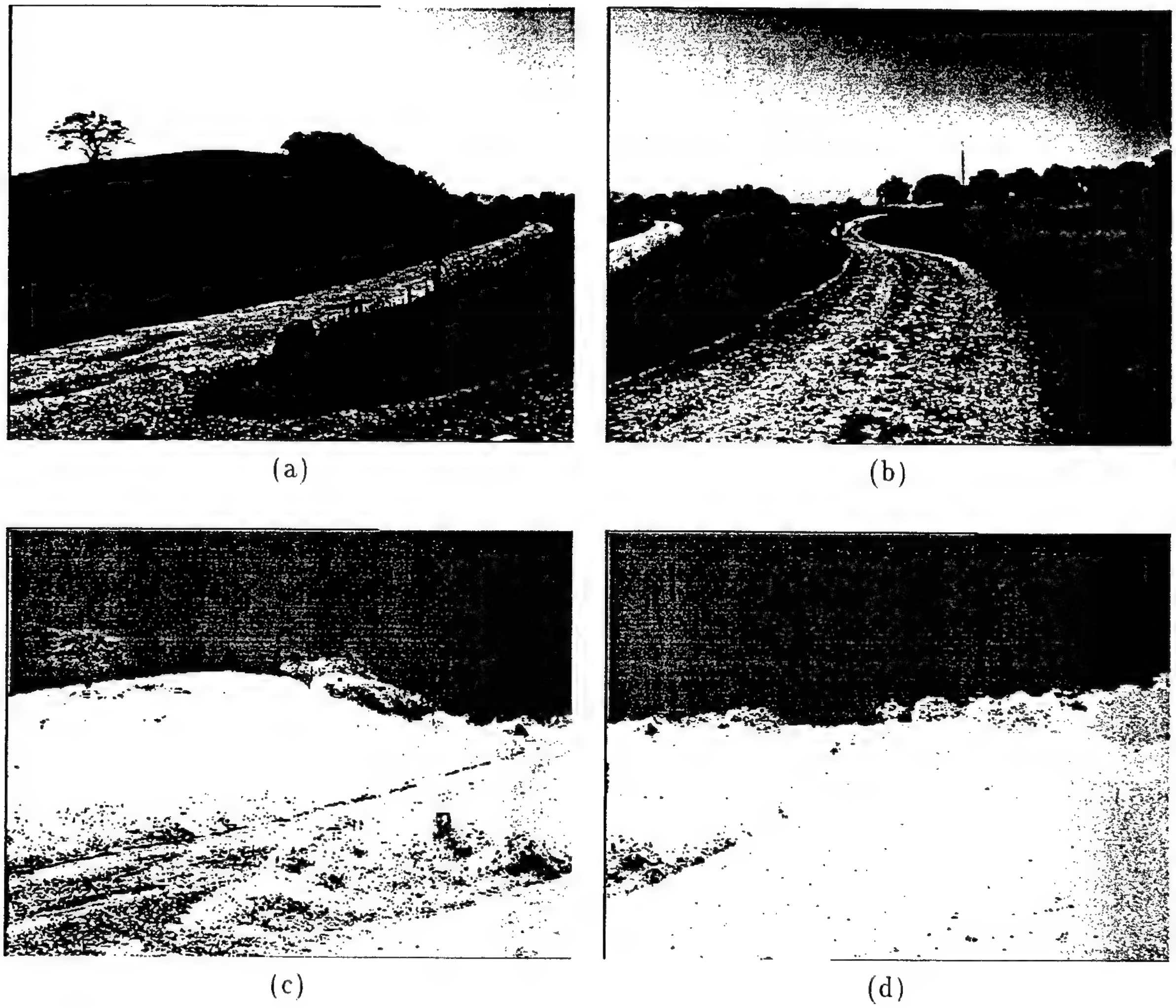
# 8 APPENDIX: A Recoloring Algorithm

The algorithm presented below accepts individual pixels from a color image of an outdoor landscape scene; each such pixel is described by an 8-bit each red, green, blue triple called c-list. The algorithm returns a corresponding 4-bit pixel which can be used to assemble a "false-color" image which has the qualitative appearance of the original color image, but in which the 16 distinct values of the four bits represent specific semantic categories. At present, only five semantic categories are used in the false-color rendering. The algorithm is representative; it has not been optimized for its intended purpose and the numerical constants embedded in the algorithm are also representative rather than definitive.

This algorithm exploits color and intensity regularities present in most normal outdoor scenes. In particular, the relative blue content and brightness of a pixel both vary according to the high-to-low scale: (cloud/snow/sky) (water/rock) (ground) (live-vegetation) (shadow).

```
(defun mf-nor (c-list)
  "color based natural object recognition:
     ID codes:
        (shadows/unknown 0) (water/rock 3) (cloud/snow/sky 7) (ground 14) (live-veg 15)"

  (prog (r g b bright dark)
        (setq bright 240)
        (setq dark    30) ;; should use shadow-threshold here when scene has shadows
        (setq r (max 1 (float (first c-list)))
              g (max 1 (float (second c-list)))
              b (max 1 (float (third c-list))))
        ;; the following classifications are sequence dependent
        (when (and (> dark r) (> dark g) (> dark b)
          (return 0)))  ;; shadow or unknown
        (when (or (> b bright)
                  (and (< 60 b) (< r b) (< g b))
                  (and (< 100         b)
                       (> (* 1.20 r)  g)
                       (> (* 1.10 b)  g)
                       (> (* 1.20 b)  r)))
           (return 7))    ;; sky or haze or cloud or snow
        (when (or (and (> 80 b) (< 2 (/ r b)) (> (* 1.2 g) r))
                  (and (> 80 b) (< 2 (/ g b)))
                  (and (< b dark) (>= (* 1.2 g) r)))
          (return 15))           ;; veg
        (when (and (> r g) (> g b) (>= .27 (/ b (+ r g b))) )
          (return 14))   ;; ground
        (when (and (> (* 1.2 g) r) (> g b) (< .27 (/ b (+ r g b))))
          (return 3))    ;; water/rock
        (return 0)  ;; unknown or shadow
        ))
```



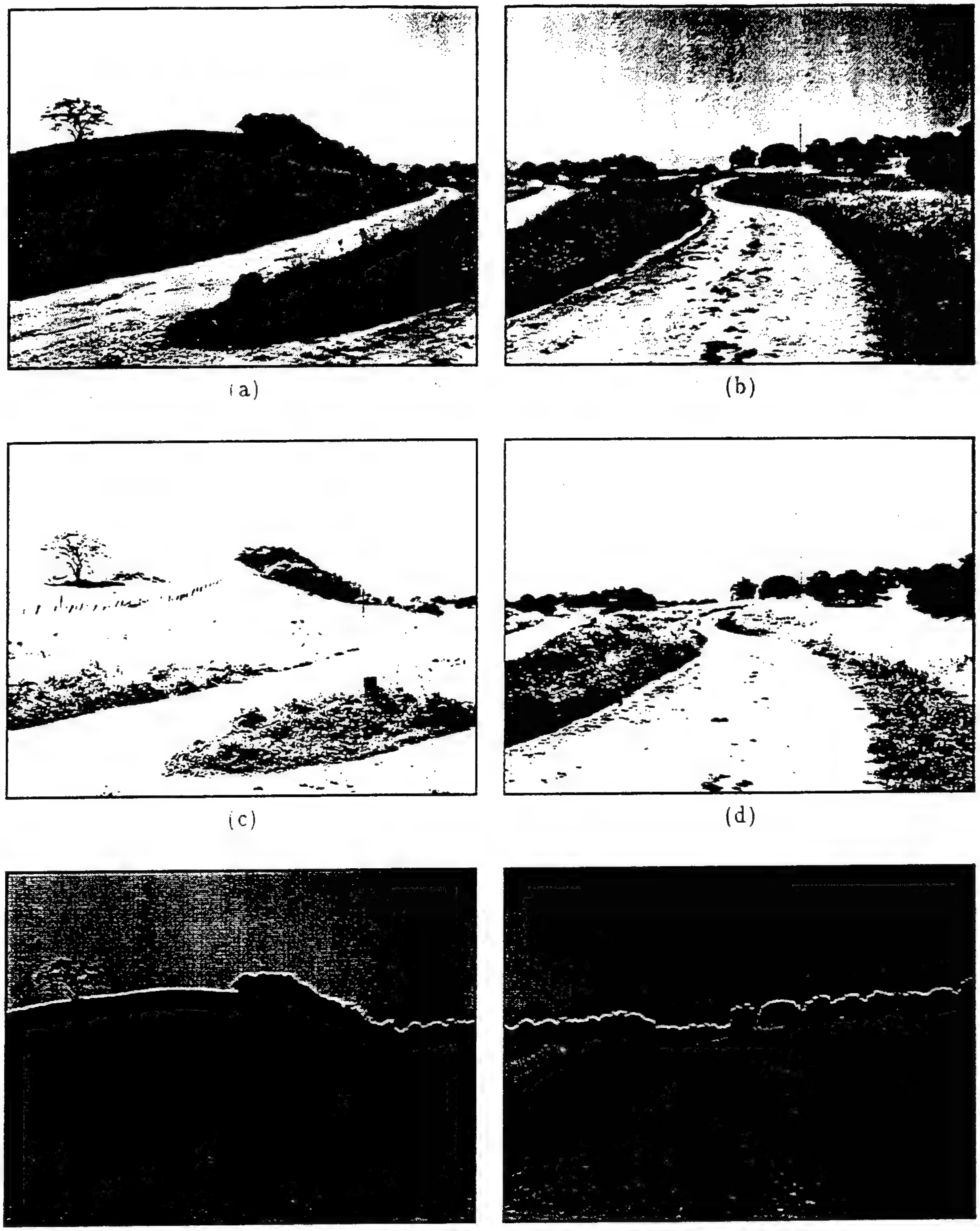
(a)

(b)

(c)

(d)

Figure 1: The Color Sketch
(a,b) Stanford Hill Scenes
(c,d) The Color Sketch (Semantic Labeling)

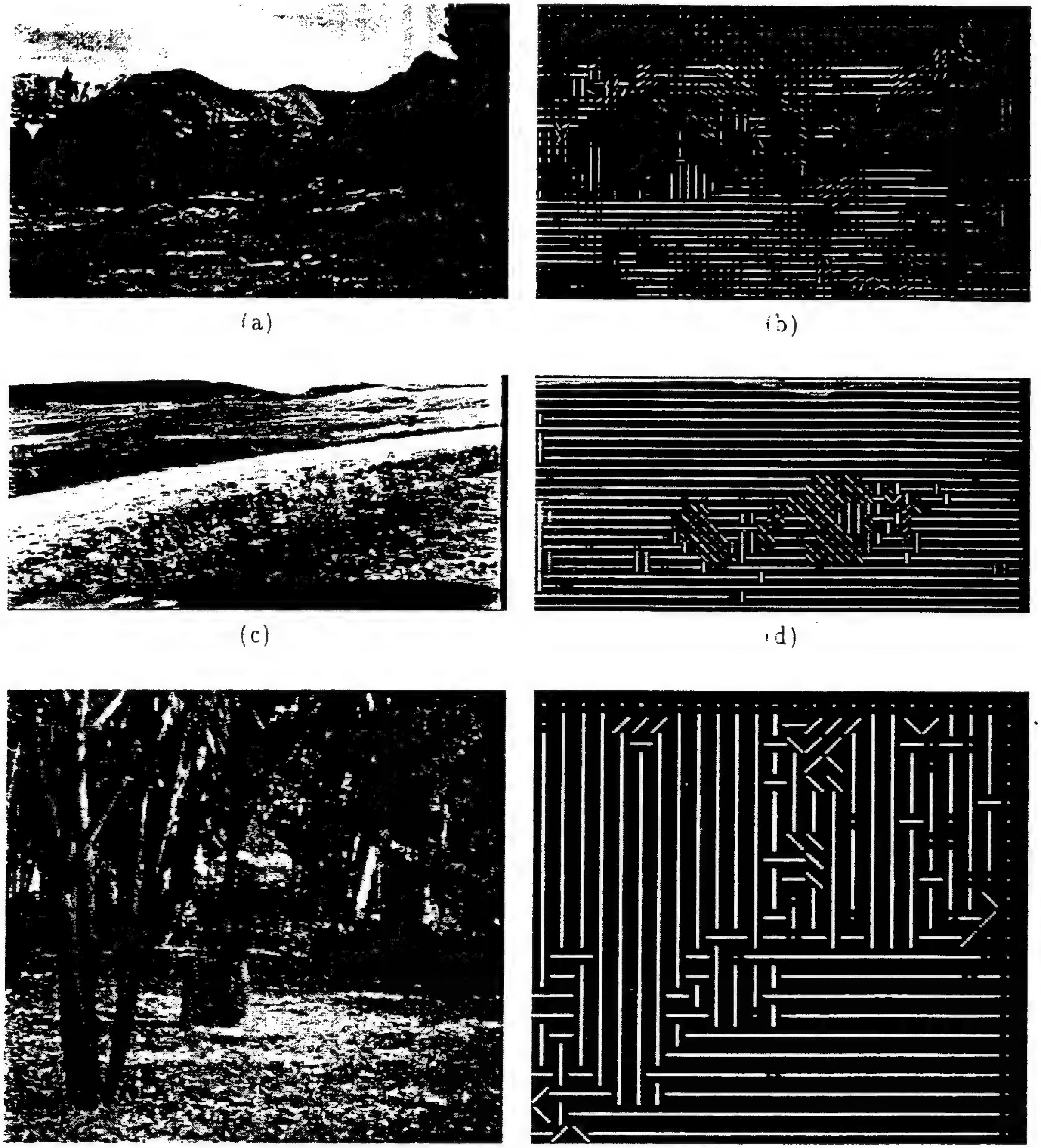


(a) (b)

(c) (d)

(e) (f)

Figure 2: Components of the Scene Sketch
(a,b) Stanford Hill Scenes
(c,d) Shadow Sketch (raised objects)
(e,f) Line Sketch (skyline)



(a)

(b)

(c)

(d)

(e)

(f)

Figure 3: The Texture Sketch: detecting raised objects (vertical striations) and horizontal surfaces (horizontal striations)

(a.b) sky. mountains. trees. meadow
(c.d) sky. hill. road. grass
(e.f) trees. grass

# APPENDIX C:

## “Sketching Natural Terrain from Uncalibrated Imagery”

Q.T. Luong

ARPA Image Understanding Workshop, May 1997

(Updated Version)

# Sketching natural terrain from uncalibrated imagery

Quang-Tuan Luong

## 1 Introduction

We have proposed a methodology to build a 3D sketch for an outdoor scene consisting of natural terrain from a pair of uncalibrated images.

There has been an extensive amount of work done in terrain reconstruction with calibrated cameras. The most successful approaches use stereo rigs or multiple views. However, there are many situations where the calibration data (camera parameters and relative position and orientation of the cameras) is not available. Although the investigation of the capacities of uncalibrated systems has become a popular research topic over the past few years there are no systems which have demonstrated the capacity to perform a usable three-dimensional reconstruction of natural terrain.

It is generally believed that the area-based approaches to stereo are the most adequate for natural terrain, since well-defined geometric features are generally lacking. These approaches produce a depth map from which further processing is necessary in order to extract higher level information about the terrain. By contrast, we propose to represent the terrain by a set of profile lines, which is the trace of the terrain surface on a plane in 3D at the given depth. This representation has a direct meaningful interpretation.

The classical approach to stereo consists in determining the disparity for each point along an epipolar line. The epipolar line is determined only by the geometry of the cameras, and within this line, each point corresponds to a different depth. By contrast, given a fixed depth, we propose to find all the points which lie at this depth. This is based on the idea that for the points which lie at a fixed depth, there is an analytical relation between their projections in multiple images. By sweeping 3D space with planes at a set of different depths, a representation of the terrain is obtained. We show that in order to generate a qualitatively useful elevation map, full calibration of the cameras is not necessary. Instead, the only requirement, in addition to the epipolar geometry, is the identification of correspondences on the horizon, a technique well adapted to the type of scenes we consider. This makes it possible to apply our technique with as few as two uncalibrated views. Second, we propose a method based on curve evolution to generate the profile lines. This makes it possible to enforce continuity, smoothness, and uniqueness constraints in space

## 2 The principle of the approach

In this section, we explain the theory behind our approach, which was established during the first phase of this project. This appeared in [2].

### 2.1 Affine calibration

Affine calibration of a pair of views consists in determining enough geometric parameters for this pair of views so that the ambiguity in reconstruction will be at most an affine transformation of space.

We use the pinhole model: the relationship between 2-D pixel coordinates and any 3-D world coordinates can be linearly described by a $3 \times 4$ matrix $\tilde{\mathbf{P}}$, which maps points from $\mathcal{P}^3$ to $\mathcal{P}^2$. Under this model, the relationship between the projective retinal coordinates of a point $\mathbf{m}$ and the projective coordinates of the corresponding epipolar line $\mathbf{l}'_m$ is linear. The *fundamental matrix* [3] describes this correspondence.

The Fundamental matrix has 7 independent parameters which represent the only generic information relating two uncalibrated views. It can be computed using only point correspondences, and this is now a classical problem. Although methods have been developed to automatically generate point correspondences, this remains a most difficult problem in computer vision.

Knowing only the fundamental matrix makes it possible to perform a 3D reconstruction up to a general projective transformation of space. Such a representation is not very useful for sketching purposes, because the degree of deformation can be very large. A way to avoid these problems is to perform an affine calibration of the pair of cameras. This means that in addition to the fundamental matrix, we need to identify the homography

$\mathbf{H}_\infty$ of the plane at infinity, defined as follows: the projective coordinates of two points $\mathbf{m}$ and $\mathbf{m}'$, projections in the first and second image of a point at infinity, are related by:

$$\mathbf{m}' \simeq \mathbf{H}_\infty \mathbf{m} \tag{1}$$

## 2.2 The profile lines

We describe how, given a pair of affinely calibrated cameras, we can represent the profile lines.

Having performed affine calibration, if we know the vanishing line $\mathbf{r}$ of a plane in the first view, we can define a set of planes $\Pi_Z$ which are parallel to this plane. No 3-D reconstruction is needed for that. Instead, the plane $\Pi_Z$ is defined by its homography $\mathbf{H}_Z$, such that the projective coordinates of two points $\mathbf{m}$ and $\mathbf{m}'$, projections in the first and second image of a point of $\Pi_Z$, are related by:

$$\mathbf{m}' \simeq \mathbf{H}_Z \mathbf{m}$$

Let us consider the family of homographies:

$$\mathbf{H}_Z \simeq \mathbf{H}_\infty + \frac{1}{Z}\mathbf{e}'\mathbf{r}^T$$

where $\mathbf{e}'$ is the epipole in the second image (projection of the optical center of the first camera). The direction of the plane $\Pi_Z$ is given by its intersection $L$ with the plane at infinity $\Pi_\infty$, a line at infinity in 3D whose projection in the first image is the vanishing line of $\Pi_Z$ in this image. Since the projections $\mathbf{m}$ of points of $L$ satisfy $\mathbf{H}_Z\mathbf{m} \simeq \mathbf{H}_\infty\mathbf{m}$, the projective equation of the vanishing line is $\mathbf{r}^T\mathbf{m} = 0$. All the planes obtained by varying $Z$ have the same vanishing line, therefore they are parallel.

Strictly speaking, a profile line is the trace in a vertical plane of the surface which represent the terrain. We can define a family of parallel vertical planes if the vertical vanishing point can be identified in the images. If the vertical direction cannot be identified reliably, then we can still apply these ideas using the family of planes which are parallel to one of the camera's retinal plane (ie fronto-parallel with respect to this camera), and obtain a reasonable approximation if this camera is about level. These planes are obtained with $\mathbf{r} = [0, 0, 1]^T$, which means that their vanishing line is the line at infinity in the image plane, ensuring that the image plane and the planes $\Pi_Z$ are parallel. Each of these planes $\Pi_Z$ is therefore defined by its homography matrix:

$$\mathbf{H}_Z \simeq \mathbf{H}_\infty - \mathbf{e}'[0, 0, \frac{1}{Z}] = \begin{bmatrix} H_{11} & H_{12} & H_{13} + \frac{1}{Z}e'_1 \\ H_{21} & H_{22} & H_{23} + \frac{1}{Z}e'_2 \\ H_{31} & H_{32} & H_{33} + \frac{1}{Z}e'_3 \end{bmatrix}$$

Although the calibration is only affine, the parameter $Z$ has a metric interpretation. It represents the perpendicular distance of the plane to the origin, up to an unknown scale factor.

Knowing the homography $\mathbf{H}_Z$ makes it possible to determine whether a point $\mathbf{m}$ in the first image is the projection of a 3D point which belongs to the plane $\Pi_Z$: if it is the case, its correspondent in the second image should be $\mathbf{H}_Z\mathbf{m}$. The profile line in the first image corresponding to the relative depth $Z$ is the set of points $\mathbf{m}$ in the first image, such that their correspondant in the second image is the point $\mathbf{H}_Z\mathbf{m}$.

# 3 Algorithmic developments

There are two distinct algorithmic issues. The first one is concerned with affine calibration of the pair of images. The second one is the localization of the profile lines. We detail the progress made on each part.

## 3.1 Affine calibration

**Determining points at infinity** In a natural scene of the type we are interested in, an appropriate method is to identify corresponding points at the horizon.

This can be done using a set of simple, but efficient heuristics, as shown by Fischler in [1], where a method to extract the skyline was described.

**Robust computation of the infinity homography** Once the Fundamental matrix is determined, there are only three degrees of freedom for the infinity homography. These three degrees of freedom can be represented by the vector $\mathbf{r}$ such that:

$$\mathbf{H}_\infty = [\mathbf{e}']_\times \mathbf{F} + \mathbf{e}'\mathbf{r}^T \tag{2}$$

where the symbol $[.]_\times$ designates the skew-symmetric matrix associated to the cross-product: for a given vector $\mathbf{x}$ the matrix $[\mathbf{x}]_\times$ is such that for any vector $\mathbf{y}$, $\mathbf{x} \times \mathbf{y} = [\mathbf{x}]_\times \mathbf{y}$.

Reciprocally, once the infinity homography is determined, there are two degrees of freedom for the Fundamental matrix, which are the two coordinates of one epipole, since the Fundamental matrix is obtained from the infinity homography by:

$$\mathbf{F} = [\mathbf{e}']_\times \mathbf{H}_\infty \tag{3}$$

We have developed three different approaches to affine calibration. Each of these approaches consist in first solving a linear least-squares system, and then using this result for a final non-linear minimization, in which the vector relevant parameters are determined by minimizing the least-squares sum of some image error terms.

1. First compute the Fundamental matrix (from all points), then the vector $\mathbf{r}$ (using the points of the horizon) thanks to Eq. (2). This was the method presented in [2]. Subsequently, we developed two other methods:
2. First compute the infinity homography (using the points of the horizon), and then the epipole $\mathbf{e}'$ (from all points) thanks to Eq. (3).
3. Simultaneous computation of the Fundamental matrix and of the three affine parameters. This is done by minimization of the geometric error function:

   $$\sum_i \{d_l(\mathbf{m}'_i, \mathbf{e}' \times \mathbf{H}\mathbf{m}_i)^2 + d_l(\mathbf{m}_i, \mathbf{H}^T(\mathbf{e}' \times \mathbf{m}))^2\} + \sum_j \{d_p(\mathbf{m}'_j, \mathbf{H}\mathbf{m}_j)^2 + d_p(\mathbf{m}_j, \mathbf{H}^{-1}\mathbf{m}'_j)^2\}$$

   over the 8 parameters of $\mathbf{H}$, the homography of the plane at infinity, and $\mathbf{e}'$, the epipole in the second image, $d_l$ being the Euclidean distance between a point and a line, and $d_p$ the Euclidean distance between two points. The correspondences $(\mathbf{m}_j, \mathbf{m}'_j)$ are points on the horizon, the other correspondences being general.

We conducted many simulations and determined that the third method gives the most consistent results.

## 3.2 Generation of the profile lines

The idea is to compute a correlation score between the point $\mathbf{m}$ in the first image and the point $\mathbf{H}_Z\mathbf{m}$ in the second image. If $\mathbf{m}$ is a projection of a point which lies on the plane $\Pi_Z$, then this correlation score should be high.

**Computation of an absolute correlation score** By computing such a score for each point of the first image, we create a correlation image, in which we expect the high values to correspond to points of the profile line. This approach was presented in [2], where a few correlation images were presented. From there the profile lines would be obtained by linking the maxima, but instead we chose to pursue the implementation differently.

Most of the problems that we encountered with this approach was due to the difficulty of obtaining reliable correlation measures. We have tried several classical scoring schemes. Experiments with several sets of images showed that these correlation measures were not adequate to match two views which are widely spaced. We began to study alternative correlation measures, but this turned out to be a difficult problem with a larger scope than this project.

**Computation of a relative correlation score** A subsequent improvement has been to detect a change of disparity sign rather than a local maximum, by performing a local search along the epipolar line, resulting in a local disparity map between the first image, and the second image warped by the homography $\mathbf{H}_Z^{-1}$. The profile line is then obtained by linking the zero-crossings.

We also tried a one-step approach consisting to warp the second image by the homography $\mathbf{H}_\infty^{-1}$ and then compute explicitly the whole disparity map, which in theory gives the all the profile lines. However, in practice we have found this approach to yield inferior results compared the previous one, because of the larger projective deformation induced on most of the image.

**Profile line linking** We have developed a snake-based approach to link the profile lines. Two sets of equidistant profile lines are shown in the same image in Fig. 1. The first set of lines is obtained with the relative depths $Z = 40, 60, 80, 100, 120$, whereas the second set is obtained with the relative depths $Z = 240, 340, 440$. The results look qualitatively correct for the first set. In particular the structure of the depression and the slope on the right is captured. For the second set, because we have not enforced ordering, in the portion of the image corresponding to the horizontal ground, the profile lines are somewhat tangled. However, it can be noticed that the profile of the tree has been captured, even though it is quite far.

**Taking into account domain constraints** In the general case, the cross-sections by a plane are sets of closed curves, rather than a single curve. However, in our case there is a simplification, which makes it possible to consider a profile line as a function $v = f(u)$, where $u$ is the horizontal axis of the image and $v$ the vertical axis: we assume that to each point (X,Y) is associated a single elevation $h(X, Y)$. This hypothesis is verified if we consider that everything which is under an overhanging object is actually part of this object. Under this hypothesis, the profile lines cannot cross. We have not yet taken advantage of this observation.

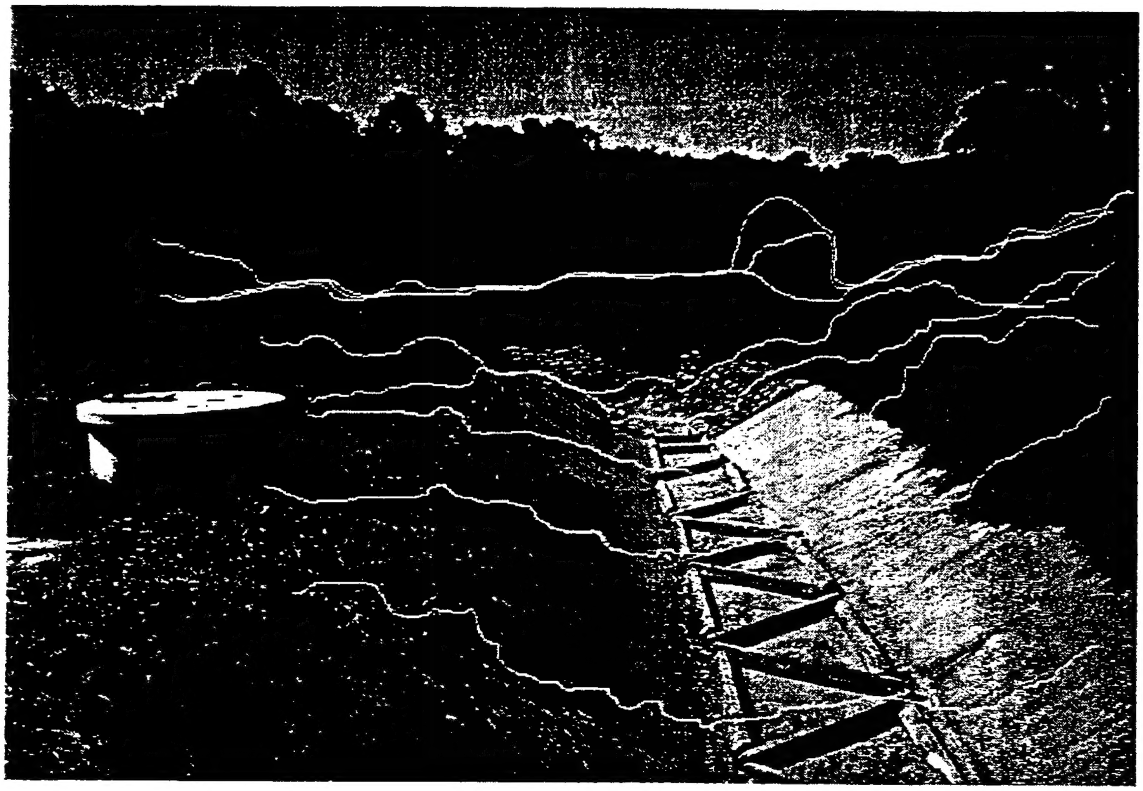


Figure 1: Two sets of profile lines superimposed on the first image. The first 5 profile lines are equidistant, and so are the last 3 profile lines in a distance.

## 4 Summary

We have proposed a scheme for sketching natural terrain. This scheme takes advantage of general domain-specific constraints: the availability of an horizon line, and the $2\frac{1}{2}$ nature of the natural terrain. By taking advantage of these constraints, we are able to propose a method which has the potential to produce a useful representation from minimal data (two uncalibrated images, one of which is approximatively level) in a domain which has been traditionally considered to be difficult. We believe that because no three dimensional reconstruction is involved, our scheme could be more stable than traditional approaches in the case these approaches would be applicable.

Our method produces a dense sketch consisting of a set of profile lines where the order with respect to the dimensions of height above the ground plane and depth are correct. These profile lines are a semantically meaningful representation of natural terrain.

The limits of applicability of the method are those of the correlation-based approaches to image-matchinge, a problem of wider scope.

## References

[1] M.A. Fischler. Robotic vision: Sketching natural scenes. In *ARPA Image Understanding Workshop*, Palm Springs, CA, 1996.

[2] Q.-T Luong. Sketching natural terrain from uncalibrated imagery. In *ARPA Image Understanding Workshop*, pages 519–528, New Orleans, LA, 1997.

[3] Q.-T. Luong and O.D. Faugeras. The fundamental matrix: theory, algorithms, and stability analysis. *Intl. Journal of Computer Vision*, 17(1):43–76, 1996.